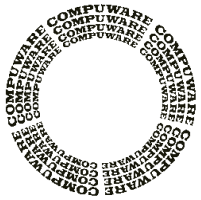
Enterprise Common Components

Installation and Customization Guide
MVS Version

Release 1.5



COMPUWARE®

Please direct questions about Enterprise Common Components
or comments on this document to:

Compuware Shared Services Technical Support
Compuware Corporation
31440 Northwestern Highway
Farmington Hills, MI 48334-2564
1-800-538-7822

Outside the USA and Canada, please contact
your local Compuware office or agent.

Doc. CWCYIX1E
July 30, 2002.

Contents

Figures

Tables

Summary of Changes

This section includes brief descriptions of enhancements made to Enterprise Common Components (ECC) 1.5 covering Compuware Shared Services (CSS) 7.9 and License Management System 2.0.

Enterprise Common Components Release 1.5

Enterprise Common Components 1.5 incorporates the Compuware Shared Services upgrade from Release 7.8 to 7.9.

Compuware Shared Services 7.9

CSS Release 7.9 offers the following new features and enhancements:

- Shared Directory Support for Abend-AID XLS Reports

  Abend-AID XLS users can now create/attach multiple DDIO report files to a Shared Directory. The Shared Directory support provides users with a single point of access to all of their report repositories.

- Abend-AID XLS Shared Directory Report Routing Rules Support

  Abend-AID XLS reports can be routed to specific subsets of report databases by specifying routing rules (see the Abend-AID XLS *Installation and Customization Guide* for details) when using the Shared Directory support structure. Based on user-supplied routing criteria (for example, Job Name/Job Accounting Info, etc.), abend reports can be directed to specific subsets of report databases to facilitate customer needs. For example, payroll job abends could be routed to a restricted subset of report databases for security reasons.

- Shared Directory Source Support for All Compiler Languages

  All compiler languages can now be used with the Shared Directory to store/retrieve their Source Listings. This allows Abend-AID XLS users to specify a single Source Listing Shared Directory name in place of a source DDIO file to retrieve a source listing, or merge a source listing with a report from a pool of source listing databases. The Shared Directory can be defined with a DBMODEL parameter (or it can be added later), which will allow CSS to dynamically allocate a new database file if all others are full.

  Note: XPEDITER/TSO and XPEDITER/CICS do not support Shared Directory processing.

- Process Compiler Listings On-The-Fly

  When viewing an Abend-AID XLS report through the Viewing Facility, CSS attempts to merge the source listing with the report. If the source listing cannot be found, the customer will have the ability to allow a listing to be compiled (preprocessor) into a source DDIO file or have an existing compiler listing post-processed into a source DDIO file. This enhancement is provided by distributing an exit to two CLISTs (or REXX EXECs) so the customer can write a dialog that will allow a listing to be compiled as described above. Sample dialogs have been supplied to assist the customer in dialog development.

- Abend-AID E-Business Name Changed to Abend-AID for WebSphere MQ

The Abend-AID E-Business product has been renamed to Abend-AID for WebSphere MQ. All references to E-Business have been replaced in the CSS Viewing Facility.

- Source Support for VisualAge PL/I Abend-AID XLS Reports

  CSS now provides support for customers who view Abend-AID XLS reports via the CSS Viewing Facility to also view VisualAge PL/I program source merged with the Abend-AID XLS report.

- Source Support for Multiple Assembler Programs on the Calling Chain List in the Viewing Facility

  In previous releases, when two or more Assembler programs were linked together, only the abending program would get source support. With this enhancement, all programs on the calling chain can be viewed with their source merged with the Abend-AID XLS report.

- Revise Printed Abend-AID XLS Report So It Matches the Viewing Facility Report

  The printed Abend-AID XLS report has been modified so the format of the report is in the same order as the Viewing Facility.

- Viewing Facility Updated to Display COBOL PIC and VALUE Information from the Working-Storage Section

  In previous releases, the COBOL PIC and VALUE information were available in the stored source listing member. The Viewing Facility has been enhanced to include this information in both the online and printed reports in addition to the stored member.

- Viewing Facility Hardcopy Print Routines Updated to Show the Report to be Processed on the Print Routing Screen

  Previously, when multiple members were selected for printing from the Viewing Facility, the report (or source listing member) was not identified on the print routing screen. This enhancement will display some identifying information each time the routing screen redisplays.

- Link to Viewing Facility From Within SDSF When Viewing an Abend-AID XLS Dump

  This enhancement enables the user to invoke the Viewing Facility and display the Abend-AID XLS report from within SDSF. The linkage mechanism will place the user directly into the proper report without the need to make note of a jobname or report number, and without having to navigate the menus. At the conclusion of the Viewing session, the user will be returned to SDSF.

- Distributed Viewing Support for CICS Abend-AID/FX Transaction Dumps

  Abend-AID XLS customers have the ability to view abend reports and source listings from DDIO files that may be on a remotely connected system. This enhancement provides support for using remote DDIO files to CICS Abend-AID/FX users.

- Cursor-Sensitive Panel Selection From Within the Viewing Facility

  This enhancement allows the user to tab to certain fields in the Viewing Facility panels, press ENTER, and be routed directly to the appropriate screen for more information on that field.

Enterprise Common Components Release 1.4

Enterprise Common Components 1.4 incorporates the Compuware Shared Services upgrade from Release 7.7 to 7.8.

Compuware Shared Services 7.8

CSS Release 7.8 offers the following new features and enhancements:

- VisualAge PL/I for OS/390 R2.2 Support

  This enhancement provides the capability to use the VisualAge PL/I for OS/390 V2.R2.24 and V2.R2.M1 compiler in the following ways:

  – Preprocess and compile PL/I source code using the new compiler.

  – Postprocess a listing previously generated by the new compiler.

  – Create a DDIO source member closely compatible with source members from earlier PL/I compilers so that it can be accessed and used by other Compuware products that currently use CSS's DDIO source format.

- Support for COBOL Report Writer

  This enhancement supports the COBOL Report Writer Precompiler, PP 5798-DYR with a related run-time library designated PP 5798-DZX. Users may now insert COBOL Report Writer statements into a COBOL program. Use of the input and/or print exit is supported by the Compuware preprocessor.

- Support for COBOL DB/2 Co-processor

  The new COBOL V2R2 compiler features a DB/2 co-processor (requires DB/2 version 7). This replaces the need to use the DB2 preprocessor to translate a COBOL listing that contains "EXEC SQL" statements. The new co-processor is invoked with the 'SQL" compiler option. The CSS COBOL language processor will accept listings compiled from this co-processor and properly create SLS DDIO members with appropriate CLIST entries for the EXEC instructions.

- Support for COBOL 2.2 COMP-5 and 31-digit Decimal Variables

  The new COBOL V2R2 compiler allows packed and zoned decimal data (COMP-5 and DISPLAY) with precision up to 31 digits when the new compiler option ARITH(EXTEND) is specified.

- Show Value of Subscripted Variables in Diagnostics Section (COBOL only)

  CSS previously referred viewers to the Working-Storage section whenever there was a subscripted variable (element of an array) in the Diagnostics section. CSS now provides the actual value of the variable whether the subscript is specified as variable, integer, or variable+/-integer. These variables are shown in the "Current values of fields on abending statement" section.

- Remove Sorts from the Assembler Language Processor

  This enhancement removes external sort processing from the CSS Assembler postprocessor and replaces it with a hashing technique. This hashing technique is similar to that used by the PL/I postprocessor, which currently performs no external sorting. Eliminating the sorts provides the following advantages:

  – Performance benefit (CPU and I/O reduction)
  – Reduces size of execution JCL (no sortlib or sort work DD statements)

- Toleration Support for the New Assembler R1.4 Features

  This enhancement allows Assembler options to be specified in an external file. It also provides for use of the two new DC types in High Level Assembler 1.4 for the new 64 bit architecture.

  – FD — double word aligned fixed-point constant
  – AD — double word aligned address constant

- Support for STOP ALL MQ Requests

  This enhancement identifies and flags the new MQSERIES command calls for testing by XPEDITER/CICS users. The existing CLIST table is modified at processing time to handle the new MQ type. CSS returns the statements containing the CALL VERB. Supported call types are: CONNX, BACK, BEGIN, DISC, CLOSE, OPEN, CMIT, CONN, PUT1, SET, XCNVC, _DATA_CONV_EXIT, GET, INQ, and PUT.

- Support for STOP ALL WEB Commands

This enhancement identifies and flags the new CICS TS 1.3 'EXEC CICS WEB typ2" API command calls for debugging WEB-based applications by XPEDITER/CICS users. The existing CLIST table is modified at processing time to handle the new WEB type. CSS returns the statements containing the CALL VERB. The supported call types are: EXTRACT, STARTBROWSE, READNEXT, ENDBROWSE, READ, RECEIVE, WRITE, SEND, and RETRIEVE.

- Reprocess Enhanced Listings

  CSS currently provides the ability to save COBOL compile listings in an "enhanced" format. This enhancement processes these enhanced listings via the CSS postprocessor and stores them in the DDIO file.

- Enhanced Source Support for BL Cells

  Current view-time support properly displays only BLL and BLW cells. As a result, customers receive many 'DATA ERROR RC=10' messages for other types of BL cells. In many cases this is caused when Abend-AID does not collect all information at abend time. For the other instances, CSS has improved its support during view-time for the different types of BL cells.

- Upgrade Compuware Base Services/HCI to 4.1

  This enhancement brings the base services used by Distributed Viewing to the same release level used by CICS Abend-AID/FX by ensuring that FX and CSS distribute the same level of Compuware Base Services/HCI. HCI must be installed to run Distributed Viewing Support.

- Upgrade to HCI 2.3

  This enhancement prevents Host Communication Interface (HCI) version mismatch errors between CSS and CICS Abend-AID/FX. It also changes the communication protocol from LU 6.2 to TCP/IP.

- Shared Directory EXAMINE Command

  Depending on the abend activity, it is possible for one or more report files to contain (or not contain) reports listed on the Shared Directory display. This utility command examines (and optionally fixes) the physical integrity of the Shared Directory, detects out-of-sync conditions, provides a report, and can (optionally) automatically synchronize the report files and Shared Directory.

- Modify SMP/E MCS to Delete Pre-existing CSS Releases

  Previously, when users wanted to install a new CSS release in the same zones as a previous CSS release, miscellaneous link-edit errors occurred because SMP/E attempted to merge the old and new releases (linking them together). This enhancement causes SMP/E to detect that a previous CSS release exists, to automatically delete the old CSS release, then to install the new release. This does not affect users who use separate zones for their CSS releases.

- Save Compiler Listing DSN on Utilities Panel

  CSS saves the name of the dataset containing compiler listings between sessions on the 'Create JCL - Postprocess Listing' utilities panel.

- Vocabulary/Appearance Changes

  The following panel changes have been introduced in this release:

  – Improved vocabulary on CSS utility screens to clarify the steps required to use source support (XLS).

  – Clearly identifies source and report 'listings' on screen prompts.

  – Changed the 'create JCL for LP preprocessing and postprocessing' prompt to more accurate terminology.

License Management System Release 2.0

License Management System Release 2.0 includes the following enhancements:

- Enforcement of new Compuware license models on z/900 logical servers executing the z/OS operating system.
- New 14-day grace period allowing Compuware products to run when a logical server exceeds licensed limit or when the name of the logical server doesn't match the name licensed.
- New 14-day grace period allowing Compuware products to run on a new or upgraded CPU that is not yet licensed.
- New/redesigned License Management System reports.

Introduction

This manual contains instructions for installing, customizing, and maintaining Enterprise Common Components (ECC).

Enterprise Common Components is essentially a packaging method for the following Compuware facilities:

- License Management System (LMS)
- Compuware Shared Services (CSS)
- Base Services/HCI — required only for Abend-AID XLS Distributed Viewing Support (DVS).

Manual Structure

This *Enterprise Common Components Installation and Customization Guide* is composed of the following chapters:

Chapter 1, "ECC Overview": A description of the individual components that make up Enterprise Common Components with overview information on each of the parts.

Chapter 2, "Preparing for ECC Installation": Critical information for planning your installation of the License Management System, Compuware Shared Services, and/or Distributed Viewing Support. This chapter also lists which components are required based on your Compuware product, and also lists the libraries created during installation with SMP/E.

Chapter 3, "Installing Enterprise Common Components": Step-by-step instructions for installing ECC.

Chapter 4, "Applying Maintenance": Mandatory step-by-step instructions for applying maintenance to your installation of ECC.

Chapter 5, "LMS Customization": Step-by-step instructions for setting up and customizing LMS for use with the Compuware products licensed by your organization.

Chapter 6, "CSS Customization": Step-by-step instructions for customizing your installation of CSS and the Compuware Viewing Facility (Compuware/VF).

Intended Audience

This manual is intended for installers and application programmers. It is only to be used for installing the MVS/ESA, OS/390, and z/OS versions of Compuware products.

Related Publications

For more information about Compuware Shared Services or License Management System, refer to the Compuware documentation shipped with the Enterprise Common Components tape. For more information about Distributed Viewing Support, refer to the *Abend-AID XLS Installation and Customization Guide*. Instructions for accessing this

documentation are given in the following paragraphs. For more information on compiling COBOL, PL/I, or Assembler programs, refer to the appropriate IBM manuals.

FrontLine Support Web Site

You can access online technical support for Compuware products via our FrontLine support web site. You can read or download documentation, frequently asked questions, and product fixes, or directly e-mail Compuware with questions or comments. To access FrontLine, you must first register and obtain a password at **http://frontline.compuware.com**.

Online Documentation

Documentation for this product is provided on CD-ROM in several electronic formats. PDF files can be viewed with the free Adobe Acrobat Reader, available at **http://www.adobe.com**. HTML files can be viewed with any standard web browser. BookManager softcopy files can be viewed with any version of IBM BookManager READ or the IBM Softcopy Reader. To learn more about BookManager or download the free Softcopy Reader, go to **http://booksrv2.raleigh.ibm.com**.

World Wide Web

Compuware's site on the World Wide Web provides information about Compuware and its products. The address is **http://www.compuware.com**.

Notation Conventions

This section describes common terms and syntax conventions used in this manual. A change bar (|) indicates an update to the manual for this release.

Terminology

The following terms are defined as they are used in this manual:

- **DDIO (Dump Dataset Input Output) file** is used generically to mean any report file for Abend-AID, any source listing file for Abend-AID XLS, CICS Abend-AID/FX, XPEDITER/CICS, and XPEDITER/TSO, any shared directory or transaction database for CICS Abend-AID/FX, and Source Listing Shared Directories and databases for Abend-AID XLS. For more information about DDIO files, refer to the *Compuware Shared Services User/Reference Guide*.
- **Member** is used to identify an individual diagnostic report in a report file, a source listing in a source listing file, or a profile in a profile file.
- **Entry** is used to identify an individual transaction diagnostic report in a transaction database for CICS Abend-AID/FX.

Reading the Syntax Diagrams

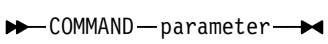
Syntax diagrams define primary command syntax.

A *parameter* is either a keyword or a variable.

- All KEYWORDs are shown in uppercase characters and must be spelled exactly as shown. You cannot substitute another value. If any part of a KEYWORD is shown in lowercase characters, that part is optional.

- Variables are user-specified values and are printed in lowercase italics. For example, *dataset-name* indicates you are to substitute a value.

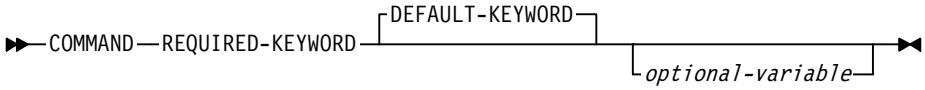
The syntax for commands is described in diagrams that help you visualize parameter use. The following example shows a command and a parameter:



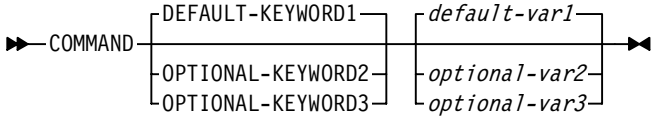
Read the diagrams from left to right and from top to bottom. These symbols help you follow the path of the syntax:

- ▶▶ indicates the beginning of a statement.
- → indicates the statement is continued on the next line.
- ► indicates the statement is continued from the previous line.
- →◀ indicates the end of a statement.

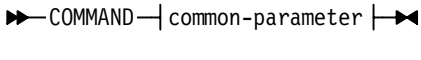
Required parameters appear on the horizontal line (the main path). Optional parameters appear below the main path. Default parameters appear above the main path and are optional. The command will execute the same whether the default parameter is included or not.



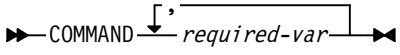
Vertically stacked parameters are mutually exclusive. If you must choose a parameter, one item of the stack appears on the main path. If the parameters are optional, the entire stack appears below the main path. If a parameter in a stack is the default, it appears above the main path.



If the same parameters are used with several commands their syntax may be documented in a separate diagram. In the command syntax, these common parameters are indicated with separators before and after the parameter name.



An arrow returning to the left indicates a repeatable item. If the arrow contains a comma, separate the repeated items with a comma.



Getting Help

Compuware continually strives to improve our software products and documentation. Feedback from our customers helps us to maintain our quality standards.

While the software should execute according to documented specifications, there are times when problems do occur. If problems arise, check the Compuware manuals for assistance. You may also need to consult your site's technical representative for Compuware assistance. Technical support is also available via our FrontLine Support web site as described in "Related Publications" on page xvii.

If you have attempted to solve the problem using the resources listed above and the difficulty persists, contact the Compuware Technical Support Hotline for assistance.

Before calling, please complete the following steps. These steps help the Technical Support staff to determine the exact cause of your problem in a timely manner.

Note: If a CSS problem is encountered when your program is being compiled, additional information may be needed by Compuware Technical Support staff. This information can be obtained using the CSS Problem Documentation Utility. Refer to Problem Documentation Utility appendix in the *Compuware Shared Services User/Reference Guide* for complete instructions.

1. Identify the ECC component (LMS, CSS, Base Services/HCI) release number and its maintenance level from the runtime and maintenance libraries.

   The release number of each component can be determined using the SxxLPTFS member, available in the installation sample library.

   Note: **xx** represents the component prefix (CS for CSS, DV for Base Services/HCI, or LM for LMS).

   This member also lists any PTFs and/or APARs that have been applied.
2. Determine the release number of the ECC-enabled product.
3. For CSS components, identify the compiler language and the release number being processed.
4. Identify the release of CICS in use if the problem involves a CICS product.
5. Identify the release of the operating system. This helps to determine operating system dependencies.
6. If an abend occurs, note the displacement and the module in which the interrupt occurs.
7. Note the sequence of issued commands that resulted in the abend, the data type, and the programming language being used.
8. Locate your Compuware customer number.

Compuware Technical Support
Compuware Corporation
31440 Northwestern Highway
Farmington Hills, MI 48334-2564
1-800-538-7822

http://frontline.compuware.com

Chapter 1.
ECC Overview

This chapter explains the composition of Compuware's Enterprise Common Components (ECC) and provides overview information on each of its component parts.

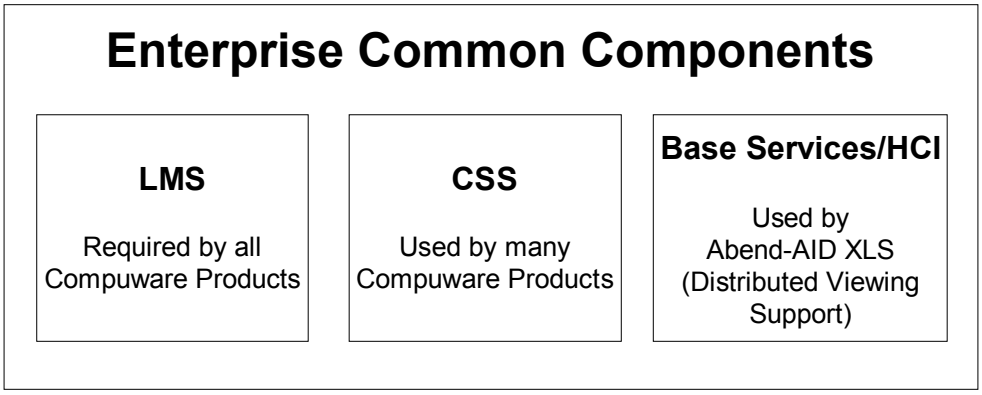
What is ECC?

Enterprise Common Components is the packaging method for the following Compuware facilities:

- License Management System (LMS)
- Compuware Shared Services (CSS)
- Base Services/HCI — Required only for Abend-AID XLS Distributed Viewing Support (DVS).

Which parts of ECC you need to install depends on which Compuware products are licensed by your organization and which parts of ECC have already been installed. All Compuware products require the License Management System. Many Compuware products utilize Compuware Shared Services to provide crucial core functionality. Base Services/HCI is used exclusively by Abend-AID XLS (Extended Language Support) Release 9.0 and more current for their Distributed Viewing Support feature. A table is provided in Chapter 2, "Preparing for ECC Installation" that makes it easy to determine exactly which parts of ECC you need to install.

Figure 1-1. ECC Composition



Each of the following components are installed using SMP/E and can be installed individually, depending on your site's requirements:

- **License Management System (LMS):** Using LMS enables you to manage access to all the Compuware products used by your organization. LMS consists of several small components which allow you to establish, maintain, diagnose, and upgrade your access to the Compuware products licensed by your enterprise:
  - An ISPF License Administration Utility application
  - A runtime environment

  - A program interface to the runtime environment employed by Compuware products.

  LMS License Certificates are English-like text files that are sent to you electronically. You will receive a License Certificate for each new product release under LMS. You can centrally manage all of your organization's License Certificates in a single License File that can then be distributed to various sites.

- **Compuware Shared Services (CSS):** An enterprise-wide tool for sharing compiled program information, CSS is utilized by many Compuware products spanning the Fault Management and Enterprise Testing Solutions product lines. For more information, see "CSS Overview" on page 1-3.
- Base Services/HCI: This component is required only for users of Abend-AID XLS Distributed Viewing Support. It allows Abend-AID XLS users to merge Abend-AID XLS reports and source listing files that are on different MVS images with unshared DASD or to access the files individually. Users access the files through the Compuware Viewing Facility. Working from individual local systems, any number of users can access reports and source listing files that reside locally and/or remotely on other MVS images. Distributed Viewing Support is available beginning with Abend-AID XLS 9.0 and more current.

LMS Overview

LMS is located on the ECC tape included in the product shipping package. LMS enables you to centrally administer Compuware's product License Certificates and manage access to Compuware products at your site. LMS includes several components that together enable you to establish, maintain, diagnose, and upgrade access to the Compuware products licensed by your enterprise. LMS replaces the customer profile utility provided with previous versions of Compuware mainframe products.

LMS Advantages

LMS provides the following advantages:

- Electronically-delivered License Certificates that contain product access parameters in a human-readable text format
- Programs and reports to verify the contents of the License File and LMS runtime environment
- Capability to employ a single centralized Enterprise License File and centralized administration of License File(s)
- Easy updates to License Certificates and the License File with no disruption to your Compuware products
- No production impact from the testing of LMS runtime systems, installation and validation of new License Certificates, or installation of new or updated LMS software
- Optional E-mail alerts and SMF logging when licensing events occur.

Operating Environment

LMS operates under IBM MVS/ESA Release 4.3 and more current, and all OS/390 and z/OS releases.

The License Management Process

The license management process begins when you acquire a Compuware product through a license agreement, trial agreement, or beta agreement.

Information Gathering

Compuware obtains the relevant information for license management from you at the time of the agreement. Your organization supplies information such as name and the sites and CPUs for which the product is being licensed. Information about the Compuware product licensed, such as product name, release number, and options licensed, is obtained from your Compuware representative.

License Certificate

Upon the completion of an agreement, Compuware creates a License Certificate representing a portion of the information in that agreement. The License Certificate is used by LMS to provide access to Compuware products. The License Certificate is **not** the same thing as a license agreement. You are still responsible for abiding by your license agreement, and although not its primary role, you will find that the LMS can help in that effort.

The License Certificate, LMS, and the Compuware product are all delivered to you by Compuware. Typically, you receive the License Certificate via e-mail, but other methods can be used, if necessary.

Installation

The first time your enterprise receives LMS, you must install the LMS software. Then you set up your LMS environment, import your License Certificate into a **License File**, activate the **Runtime License Management System**, and install your Compuware product. From then on, access to your Compuware mainframe product(s) occurs transparently.

After you accomplish the initial LMS installation, you would typically only need to import a new product License Certificate and re-initialize the LMS runtime environment in conjunction with events such as the following:

- obtaining a new release of a product under the terms of your software maintenance agreement
- adding new products through additional license agreements
- amending your agreement to include new options
- changing the CPUs licensed in your original agreement.

CSS Overview

Compuware Shared Services (CSS) is an integral component of many Compuware products spanning the Fault Diagnosis, Interactive Analysis and Debugging, and File and Data Management product lines. CSS provides storage, retrieval, and maintenance for Abend-AID reports, source listings, and transaction reports on datasets called DDIO files. It also provides language processing support for COBOL, PL/I, Assembler, and C for OS/390 (also C for MVS/ESA V3 R2).

The following products currently use CSS:

- Abend-AID XLS (Visual Age for PL/I and Enterprise PL/I language support is also available)
- Abend-AID Fault Manager
- Abend-AID for WebSphere MQ
- CICS Abend-AID/FX
- Strobe
- XPEDITER/CICS (VisualAge for PL/I and Enterprise PL/I language support is also available)
- XPEDITER/Code Coverage

- XPEDITER/TSO and XPEDITER/IMS (VisualAge for PL/I and Enterprise PL/I language support is also available)
-

Benefits of CSS

Prior to CSS Release 1.0, the components that make up CSS were distributed within each product library. For customers with multiple Compuware products, the result was a high degree of redundancy within their product libraries and a complex system of product maintenance. By combining components used by several products into a single unit, CSS provides:

- **Easier Installation**

  All CSS components can be installed at one time rather than piecemeal during installation of other Compuware products.

- **Simplified Maintenance**

  All maintenance to CSS components is applied to ONE set of libraries. It is no longer necessary to search through each product library, determine if a fix must be applied, and then apply the same fix to multiple libraries.

- **Frequent New Release Updates**

  New release updates are made available more frequently.

- **Disk Storage**

  CSS programs and subroutines reside in a separate library. Compuware customers with multiple products installed can recover disk space previously used by duplicate modules and programs using CSS.

- **Fewer JCL Changes**

  The need to change CICS startup JCL, TSO logon PROCs, compile PROCs, and/or rundecks is reduced or eliminated because a central library exists for all products using CSS. This helps to simplify installation.

- **Separate Documentation**

  Documentation for common components is no longer imbedded in the individual product manuals. In addition to your product documentation, you also receive the *Enterprise Common Components Installation and Customization Guide* in printed form, and the *Compuware Shared Services User/Reference Guide* and *Enterprise Common Components Messages and Codes* on CD. You no longer need to hunt through the product manuals for information on formatting DDIO files, printing reports and listings, and setting the parameters to run the language processors. Additionally, this ECC documentation set provides greater detail on processing options and customization than was previously available.

System Environment

CSS supports CICS Releases 1.7 through 4.1. It also supports CICS Transaction Server for OS/390 Release 1.3 and earlier and all releases of IBM MVS, as well as ISPF/PDF Versions 2.0 or more current.

The following is a list of compilers supported by the language processors:

- Assembler
  - High Level (HLASM) Assembler 1.1, 1.2, 1.3, 1.4
  - Assembler H
  - Assembler F
- COBOL

  - Enterprise COBOL for z/OS and OS/390 Release 3.1.0
  - COBOL for OS/390 and VM 2.1, 2.1.1, and 2.2 (MLE Support)
  - COBOL for MVS and VM 1.2 and 1.2.2 (MLE Support)
  - AD/Cycle COBOL/370 1.0, 1.1
  - VS/COBOL II 1.2, 1.3, 1.4
  - OS/VS COBOL 2.4

    Note: Compuware supports the MVS portion of the IBM compilers COBOL for MVS and VM, and COBOL for OS/390 and VM.

  - CA-OPTIMIZERs
    - CA-OPTIMIZER II 1.1, 1.2, 1.3, 2.0, 2.1, 2.2
    - CA OPTIMIZER III 5.0, 5.1, 6.0, 6.1

    Note: CA-OPTIMIZER II Release 2.0 supports COBOL/370.

- PL/I
  - Enterprise PL/I R3.1
  - VisualAge for PL/I R2.2
  - PL/I for MVS and VM 1.1.1
  - AD/Cycle PL/I 1.1
  - PL/I 1.5.1, 2.2, 2.3

  Note: Compuware supports the MVS portion of the IBM compilers PL/I for MVS and VM 1.1.1.

- C
  - C for MVS/ESA V3 R2
  - OS/390 C releases through V2 R10

CSS Components

CSS consists of the following components:

- Compuware common files and utilities
  - DDIO files, Shared Directories and associated Source Listing Databases
  - Batch file utilities: CWDDSUTL, CWFXSDUT, CWAASDUT, and CWDDLPUT
- Compuware Viewing Facility (Compuware/VF)
- Compuware Language Processors (COBOL, Assembler, PL/I, C, VisualAge for PL/I)
- Security Exit Program

Common Files and Utilities

CSS components include common files and utilities that provide storage, retrieval, and maintenance functions.

DDIO Files

A proprietary file access method, called DDIO, that stores Abend-AID XLS reports, transaction reports, and source listings for several Compuware products. DDIO files can be allocated as either VSAM or sequential datasets. Different products may require different types of DDIO files.

For detailed information on allocating and formatting DDIO files, see the "Allocating and Formatting DDIO Files" chapter in the *Compuware Shared Services User/Reference Guide*.

CSS Utilities and Batch File Utilities

CSS Utilities walks you step-by-step through CSS language processor functions and DDIO file manipulation using easy-to-use panels.

The batch file utilities (CWDDSUTL, CWFXSDUT, and CWDDLPUT) provide the same functions as CSS Utilities in a batch environment.

For detailed information on these utilities, refer to the "CSS Utilities", "Batch File Utility CWDDSUTL", "Batch File Utility CWFXSDUT", Batch File Utility CWAASDUT, and "Batch File Utility CWDDLPUT" chapters in the *Compuware Shared Services User/Reference Guide*.

Compuware Viewing Facility (Compuware/VF)

The Compuware Viewing Facility (Compuware/VF) provides immediate, online, menu-driven access to reports and source listings. From Compuware/VF you can view a directory that summarizes the members in a DDIO file. You can then access the source listing or Abend-AID XLS report from the directory. While viewing an Abend-AID XLS report, you can directly access any available section of the report, search for particular date or currency formats within the report, or link to Compuware File-AID.

Note: To access File-AID from Compuware/VF, one of the following File-AID releases must be installed: File-AID/MVS 8.5 or more current, File-AID for DB2 V3R9 or more current, or File-AID for IMS 4.7 with product update F1. In addition to one of the File-AID products, Abend-AID XLS 9.1 or more current must also be installed.

Compuware/VF has many features that enable you to view DDIO files easily and quickly. Its navigation commands let you directly access desired report sections to eliminate cumbersome scrolling. From Compuware/VF you can view, print, lock, unlock, and delete members in a DDIO file online. The members can easily be manually locked to ensure that they remain stored on the DDIO file until you unlock them. Compuware/VF also lets you print members at view time.

Compuware/VF allows access to the following types of DDIO files:

- Abend-AID XLS reports
- Abend-AID XLS, CICS Abend-AID/FX, XPEDITER/TSO, and XPEDITER/CICS source listings.

  Note: Viewing access to CICS Abend-AID/FX source listing files is also provided by the CICS Abend-AID/FX server. CICS Abend-AID/FX users do not need to install Compuware/VF if a server has already been configured to support viewing access. Compuware/VF may be installed, however, in order to view other products' source listing files.

For more information on the Viewing Facility, refer to the "Compuware Viewing Facility" chapter in the *Compuware Shared Services User/Reference Guide*.

Base Services/HCI Overview

Base Services/HCI is required to use Abend-AID XLS Distributed Viewing Support (DVS). Abend-AID XLS users can use DVS to view reports whether residing on a local or remote system.

Abend-AID XLS users can view both merged and base reports where either the base report, listing files, or both reside on a remote system. Access to the files is provided through the optional Distributed Viewing Support component.

With Distributed Viewing Support, Abend-AID XLS users can utilize Extended Language Support (XLS) across remote MVS systems that do not share DASD. Working from

individual local systems, any number of users can access report and source listing files that reside locally and/or remotely on other MVS images.

Distributed Viewing Support allows access to the following types of DDIO files:

- Abend-AID XLS reports
- Abend-AID XLS, CICS Abend-AID/FX, XPEDITER/TSO, and XPEDITER/CICS source listings.

Base Services/HCI is distributed with CSS on the Enterprise Common Components distribution tape. Instructions for configuring Distributed Viewing Support and setting up the servers are provided in the Abend-AID XLS product documentation. Instructions for installing Base Services/HCI can be found in the *ECC Installation and Customization Guide*.

Distributed Viewing Support is available beginning with Abend-AID XLS Release 9.0.

Note: Viewing access to CICS Abend-AID/FX source listing files is also provided by the CICS Abend-AID/FX server. CICS Abend-AID/FX users do not need to install Compuware/VF if a server has already been configured to support viewing access. Compuware/VF may be installed, however, in order to view other products' source listing files.

CSS Language Processors (LP)

The language processors capture information about a compiler listing and store it in a DDIO file. The majority of this information is gathered from the compiler listing, but optionally, SYSIN and SYSLIB are also examined.

A language processor can be run using two methods:

- preprocessor
- postprocessor

You may use the preprocessor or postprocessor (whichever is best for your site as needed.) For information about the pre- and postprocessors, refer to "Preprocessor" on page 2-10 and "Postprocessor" on page 2-11 or refer to the appropriate language processor chapter in the Compuware Shared Services User/Reference Guide.

Language Processor Types

CSS provides language processing support for the following languages:

- COBOL
- PL/I, VisualAge for PL/I, Enterprise PL/I
- Assembler
- C

The COBOL language processor reads the compiler listing produced by the COBOL compiler and writes either a compiler listing or an enhanced listing to the DDIO file. The enhanced listing merges the information normally found in the data division map (DMAP) and the condensed listing (CLIST) or procedure map (PMAP) into the source statement lines. The COBOL compiler options in effect are sorted alphabetically and appear at the beginning of the enhanced listing. The enhanced listing provides you with a condensed hardcopy of the compiler information.

The COBOL, Assembler, and PL/I listings may be used to provide source support during a debugging or diagnostic session using any of the following products:

- Abend-AID XLS
- CICS Abend-AID/FX
- XPEDITER/CICS
- XPEDITER/TSO

The C language processor reads the compiler listing produced by the C compiler and writes the compiler listing to the DDIO file and SYSCPRT. The source listing in DDIO can then be used to provide source support during a debugging or diagnostic session using:

- XPEDITER/TSO

Language support for Enterprise PL/I and Visual Age for PL/I is provided for the following products:

- Abend-AID XLS
- XPEDITER/CICS
- XPEDITER/TSO

Security Exit Program

CSS also includes the Compuware Security Exit Program. This is an optional user-written exit for your Compuware products. It can be used in conjunction with your existing security package to secure sensitive data used by the following Compuware products:

- Abend-AID XLS
- CICS Abend-AID/FX
- XPEDITER/CICS
- XPEDITER/TSO

The Compuware Security Exit allows you to control access to DDIO file members or restrict access to various commands through a user-written exit program. For the CICS Abend-AID/FX product, it lets you restrict access to some of the CWFXSDUT functions for transaction database and source listing files. For transaction databases, you can use the Security Exit Program to restrict access to individual dumps within a transaction report database.

If you install a Security Exit program, it will be called by all Compuware products installed at your site. For more information on the Security Exit Program, see the "CSS Security Exit" chapter in the *Compuware Shared Services User/Reference Guide*.

Executing CSS

CSS is executed at the following times:

- Abend-AID XLS and CICS Abend-AID/FX
  - at compile time (when source support is used)
  - at time of abend or application program failure
  - at view time when using Compuware/VF or the CICS Abend-AID/FX Viewer
  - at installation and maintenance time when formatting a DDIO file.
  - when a compile listing that has been stored elsewhere is needed for Abend-AID or CICS Abend-AID/FX viewing with source support.
- XPEDITER/TSO and XPEDITER/CICS
  - at compile time (when source support is used)
  - at execution time for interactive debugging
  - at view time when using Compuware/VF
  - at installation and maintenance time when formatting a DDIO file.

Chapter 2.
Preparing for ECC Installation

This chapter contains important information that can help you prepare to install the Enterprise Common Components (ECC) tape in the most efficient manner. Review it before beginning the installation process.

Required Components and Procedures

This section explains how to determine which parts of ECC need to be installed at your site and which chapters in this manual contain the applicable installation and customization procedures.

Component Selection

Different Compuware products utilize different portions of ECC. Use Table 2-1 to determine which parts of ECC are needed at your site.

Notice that Compuware's License Management System (LMS) is required for every product, while Base Services/HCI (which is required for Abend-AID XLS Distributed Viewing Support) is an optional component. DVS can be used with Abend-AID XLS Release 9.0 and more current.

Table 2-1. Components Utilized by Each Compuware Product

Compuware Product	LMS	CSS	Base Services/HCI
Abend-AID XLS	x	x	x *
Abend-AID Fault Manager	x	x	
CICS Abend-AID/FX	x	x	
DBA-XPERT for DB2	x		
File-AID/Data Solutions	x		
File-AID for DB2	x		
File-AID for IMS	x		
File-AID/MVS	x		
File-AID/RDX	x		
*QA*Batch	x		
*QA*Hiperstation	x		
*QA*Playback	x		
Strobe	x	x	
XPEDITER/CICS	x	x	
XPEDITER/Code Coverage	x	x	
XPEDITER/TSO and IMS	x	x	
XPEDITER/Xchange	x		
* Base Services/HCI is required only if utilizing Abend-AID XLS Distributed Viewing Support.			

Required Procedures

Determining which chapters to use in this manual is actually very simple:

- Every mainframe Compuware product utilizes LMS. Therefore, every installation of a Compuware product requires, at a minimum, that the latest maintenance be applied to LMS. So regardless of which product you are installing, you must perform the applicable procedures in the following chapters:
  - **Chapter 3, "Installing Enterprise Common Components"** (if current release has not been previously installed)
  - **Chapter 4, "Applying Maintenance"**
  - **Chapter 5, "LMS Customization"**.
- If the Compuware product you are installing requires CSS, you must also perform the procedures in **Chapter 6, "CSS Customization"**. To configure Distributed Viewing Support, see the Abend-AID XLS Installation and Customization Guide.

The information within each chapter will direct you as to which steps apply in your situation.

ECC Packaging

The ECC product tape, which includes LMS, CSS, and Base Services/HCI is packaged for installation using System Modification Program Extended (SMP/E). SMP/E is a method developed by IBM to install software products in the MVS environment.

SMP/E provides enhanced installation management and maintenance tracking capabilities. To facilitate SMP/E, the LMS, CSS, and Base Services/HCI components are shipped on the ECC tape separately from the products. The separate ECC tape contains both the base code and all cumulative, approved maintenance to-date. The ECC tape is shipped with each Compuware product that utilizes the CSS, Base Services/HCI, or License Management components. To install the ECC tape, refer to **Chapter 3, "Installing Enterprise Common Components"**.

A separate tape containing **only** the cumulative ECC maintenance is available on request. You can obtain the maintenance tape by calling the Compuware Technical Support Hotline (1-800-538-7822). This maintenance tape contains all maintenance approved to-date. To install the ECC maintenance tape to an existing installation, you only need to perform the steps described in **Chapter 4, "Applying Maintenance"**. The separate maintenance tape allows you to receive upgrades quickly — without installing a new release. It is recommended that you periodically request and apply the maintenance tape to keep your product version current.

The installation utilizes the Installation JCL Customization Facility consisting of a set of REXX EXECs, JCL skeletons, and panels. This facility prompts you for the installation information and then builds the necessary jobs required to perform the SMP/E installation. You can also view reporting information from the installation facility, such as parameters and execution information.

SMP/E Installation and Maintenance

This section describes allocating SMP/E zones, SMP/E jobs to be executed, and libraries created during the SMP/E installation.

Allocating SMP/E Zones

SMP/E uses a VSAM cluster called a CSI (Consolidated Software Inventory) to control all product installation and maintenance information. Within a CSI, there are a minimum of 3 zones:

- **Global** zone
- **Target** zone
- **Distribution** zone

If a CSI is shared among several different products, there will be a single Global zone and a separate set of Target and Distribution zones for each product installed as illustrated in Table 2-2. A site may have many CSIs, each with its own Global, Target, and Distribution zones.

CAUTION:

To prevent possible conflicts, Compuware strongly recommends that all Compuware products and components distributed via SMP/E share a single Global zone reserved solely for Compuware products. Target and distribution zones should not be shared with other products or different releases of the same product.

If you need to maintain more than one release of CSS at a time under SMP/E control, then they must be installed into separate Target and Distribution Zones. If you try to install CSS into the same Target and Distribution Zones as an existing release, then the previous release of CSS will be deleted.

Table 2-2. SMP/E Zones

Global Zone	
Target Zone Product 1	Distribution Zone Product 1
Target Zone Product 2	Distribution Zone Product 2
Target Zone Product 3	Distribution Zone Product 3

There are many different ways to allocate the zones. These methods include the following:

- The Global, Target, and Distribution zones may all reside together in a **single** VSAM cluster.
- The zones may each reside in **separate** VSAM clusters.
- The zones may each reside in a **combination** of VSAM clusters.

For example, you could have a single VSAM cluster for your Global zone and combine all the Target and Distribution zones in a second VSAM cluster. Or each set of Target and Distribution zones may be combined in a single VSAM cluster. Compuware recommends that all zones be placed in separate VSAM clusters so that each zone can be backed up and restored individually, if necessary, without impacting any other zone.

Installing with SMP/E

The SMP/E portion of the installation process for LMS, CSS, and/or Base Services/HCI consists of four basic steps:

1. Allocate and initialize the SMP/E datasets
2. Receive base and maintenance
3. Apply base and maintenance
4. Accept base and maintenance

After a function or maintenance (such as a PTF) has gone through the ACCEPT step, that becomes your current release-base level. At any time prior to the ACCEPT step, you may issue a RESTORE command to reset your Target libraries to the way they were as of the last ACCEPT command. The RESTORE command is used to remove maintenance, or even entire functions, while the ACCEPT command effectively sets a new base level of the release to which further maintenance will be applied. For more information about using the RESTORE command, refer to your site's IBM SMP/E documentation.

Installing Enterprise Common Components Using JCL Customization Facility

The Installation JCL Customization Facility consists of a set of REXX EXECs, JCL skeletons, messages, and panels contained in the SMP/E installation sample library. The facility will display several panels to obtain installation information, and then will build the necessary jobs required to perform the SMP/E installation. The facility allows you to input SMP/E information, such as the Global SMP/E and SMP/E dataset high-level qualifiers, one time. They can then be used in all SMP/E installation jobs.

Note: Compuware recommends a minimum of ISPF/PDF version 3.5 when using the Installation JCL Customization Facility. REXX is also required.

Using the Installation JCL Customization Facility, you can install any or all of the following:

- License Management System (LMS)
- Compuware Shared Services (CSS)
- Base Services/HCI — required only for Abend-AID XLS Distributed Viewing Support (DVS)

Please note the following:

- If you use Abend-AID XLS 9.0 and above and plan to use Distributed Viewing Support, you must install both the CSS and Base Services/HCI components.
- If you require only CSS, install only the CSS component.
- If you need to install more than one component, Compuware recommends simultaneously installing all components.

ECC Component Prefixes and FMIDs

Compuware has registered the following element prefixes with IBM for the ECC components:

- **LMS** for LMS
- **LCX** for CSS
- **KMP** for Base Services/HCI
- **KCW** a generic Compuware prefix.

The FMIDs for each component are listed below.

- MLMSnnn - LMS base code
- MLCXnnn - CSS base code
- NLCXnnn - CSS Japanese support
- MKMPnnn - Base Services/HCI base code
- OKMPnnn - Base Services/HCI Japanese support.

Libraries Created During SMP/E Installation

The chart in Table 2-3 lists the libraries created during the ECC installation using SMP/E.

Table 2-3. Libraries Created During Installation With SMP/E

DDname	Library Type and Content	Dataset Name as Distributed
INSTLIB	ECC SMP/E Sample Library	COMPWARE.KCWnnn.INSTALL
SMPCSI	Compuware Global CSI	COMPWARE.GLOBAL.CSI
SMPLOG	Compuware SMP/E System File	COMPWARE.GLOBAL.SMPLOG
SMPPTS	Compuware SMP/E System File	COMPWARE.GLOBAL.SMPPTS
ALMSSAMP	LMS Distribution Sample Library	COMPWARE.MLMSnnn.ALMSSAMP
ALMSLOAD	LMS Distribution Load Library	COMPWARE.MLMSnnn.ALMSLOAD
ALMSCNTL	LMS Distribution CLIST Library	COMPWARE.MLMSnnn.ALMSCNTL
ALMSMENU	LMS Distribution Messages	COMPWARE.MLMSnnn.ALMSMENU
ALMSPENU	LMS Distribution Panels	COMPWARE.MLMSnnn.ALMSPENU
ALMSSENU	LMS Distribution ISPF Skeletons	COMPWARE.MLMSnnn.ALMSSENU
LMSnnnD	LMS Distribution Zone	COMPWARE.MLMSnnn.DZONE.CSI
LMSnnnT	LMS Target Zone	COMPWARE.MLMSnnn.TZONE.CSI
SLMSSAMP	LMS Target Sample Library	COMPWARE.MLMSnnn.SLMSSAMP
SLMSLOAD	LMS Target Load Library	COMPWARE.MLMSnnn.SLMSLOAD
SLMSCNTL	LMS Target CLIST Library	COMPWARE.MLMSnnn.SLMSCNTL
SLMSMENU	LMS Target Messages	COMPWARE.MLMSnnn.SLMSMENU
SLMSPENU	LMS Target Panels	COMPWARE.MLMSnnn.SLMSPENU
SLMSSENU	LMS Target ISPF Skeletons	COMPWARE.MLMSnnn.SLMSSENU
SMPMTS	LMS SMP/E System File	COMPWARE.MLMSnnn.SMPMTS
SMPSCDS	LMS SMP/E System File	COMPWARE.MLMSnnn.SMPSCDS
SMPSTS	LMS SMP/E System File	COMPWARE.MLMSnnn.SMPSTS
ALCXCNTL	CSS Distribution Sample Library	COMPWARE.MLCXnnn.ALCXCNTL
ALCXEXEC	Distribution REXX Library	COMPWARE.MLCXnnn.ALCXEXEC
ALCXLOAD	CSS Distribution Load Library	COMPWARE.MLCXnnn.ALCXLOAD
ALCXMENU	CSS Distribution English Messages (ISPMLIB)	COMPWARE.MLCXnnn.ALCXMENU
ALCXMJPN[1]	CSS Distribution Japanese Messages (ISPMLIB)	COMPWARE.MLCXnnn.ALCXMJPN
ALCXPENU	CSS Distribution English Panels (ISPPLIB)	COMPWARE.MLCXnnn.ALCXPENU
ALCXPJPN[1]	CSS Distribution Japanese Panels (ISPPLIB)	COMPWARE.MLCXnnn.ALCXPJPN
LCXnnnD	CSS Distribution Zone	COMPWARE.MLCXnnn.DZONE.CSI
LCXnnnT	CSS Target Zone	COMPWARE.MLCXnnn.TZONE.CSI
SLCXCNTL	CSS Target Sample Library	COMPWARE.MLCXnnn.SLCXCNTL
SLCXEXEC	CSS Target REXX Library	COMPWARE.MLCXnnn.SLCXEXEC
SLCXLOAD	CSS Target Load Library	COMPWARE.MLCXnnn.SLCXLOAD
SLCXMENU	CSS Target English Messages (ISPMLIB)	COMPWARE.MLCXnnn.SLCXMENU
SLCXMJPN[1]	CSS Target Japanese Messages (ISPMLIB)	COMPWARE.MLCXnnn.SLCXMJPN
SLCXPENU	CSS Target English Panels (ISPPLIB)	COMPWARE.MLCXnnn.SLCXPENU

Table 2-3. Libraries Created During Installation With SMP/E

DDname	Library Type and Content	Dataset Name as Distributed
SLCXPJPN[1]	CSS Target Japanese Panels (ISPPLIB)	COMPWARE.MLCXnnn.SLCXPJPN
SMPMTS	CSS SMP/E System File	COMPWARE.MCXnnn.SMPMTS
SMPSCDS	CSS SMP/E System File	COMPWARE.MLCXnnn.SMPSCDS
SMPSTS	CSS SMP/E System File	COMPWARE.MLCXnnn.SMPSTS
AKMPCNTL	Base Services/HCI Distribution Sample Library	COMPWARE.MKMPnnn.AKMPCNTL
AKMPLOAD	Base Services/HCI Distribution Load Library	COMPWARE.MKMPnnn.AKMPLOAD
KMPnnnD	Base Services/HCI Distribution Zone	COMPWARE.MKMPnnn.DZONE.CSI
KMPnnnT	Base Services/HCI Target Zone	COMPWARE.MKMPnnn.TZONE.CSI
SKMPAUTH[2]	Base Services/HCI Target APF-Authorized Load Library	COMPWARE.MKMPnnn.SKMPAUTH
SKMPCNTL	Base Services/HCI Target Sample Library	COMPWARE.MKMPnnn.SKMPCNTL
SKMPLOAD[2]	Base Services/HCI Target Load Library	COMPWARE.MKMPnnn.SKMPLOAD
SMPMTS	Base Services/HCI SMP/E System File	COMPWARE.MKMPnnn.SMPMTS
SMPSCDS	Base Services/HCI SMP/E System File	COMPWARE.MKMPnnn.SMPSCDS
SMPSTS	Base Services/HCI SMP/E System File	COMPWARE.MKMPnnn.SMPSTS

[1]These datasets are part of the Japanese Language support and are only installed if Japanese Language support is required.
[2] For more information, see the Abend-AID Installation Guide.

Note: The following dataset qualifiers are used in this table:

- KCWnnn is a dummy FMID used to reserve space on the installation tape.
- MLMSnnn specifies the LMS release, where nnn is the LMS release number. For example, MLMS101 indicates LMS 1.0.1.
- LCXnnn specifies the CSS release, where nnn is the CSS release number. For example, LCX760 indicates CSS 7.6.
- MKMPnnn specifies the Base Services/HCI release, where nnn is the release number. For example, MKMP410 indicates Base Services/HCI 4.1.0.

Note: KCWnnn is a dummy FMID used to reserve space on the installation tape for miscellaneous files used during installation and customization and for the maintenance files. This FMID (KCW001 for ECC, but may vary for other Compuware SMP/E-installed products) is not intended to be installed, but merely defines some reserved file space to SMP/E.

SMP/E Tape Layout

Refer to Table 2-4 for a list of the SMP/E files on the ECC distribution tape.

Table 2-4. ECC SMP/E Tape Layout

File Number	File Name on Tape	File Contents
1	CW.COMPWARE.SMPMCS	MCS Statements
2	COMPWARE.LKCW001.F1	SMP/E Installation Sample Library
3	COMPWARE.LKCW001.F2	PTFs/APARs for this release (cumulative maintenance)
4	COMPWARE.LKCW001.F3	HOLDDATA for maintenance
5	COMPWARE.LKCW001.F4	Dummy
6	COMPWARE.LKCW001.F5	PDS containing PTF abstracts
7	COMPWARE.MLCXnnn.F1	JCLIN statements

Table 2-4. ECC SMP/E Tape Layout

File Number	File Name on Tape	File Contents
8	COMPWARE.MLCXnnn.F2	Object Modules
9	COMPWARE.MLCXnnn.F3	Panels and SAMPENU members
10	COMPWARE.NLCXnnn.F1	Japanese panels and messages
11	COMPWARE.MKMPnnn.F1	JCLIN
12	COMPWARE.MKMPnnn.F2	Object Modules
13	COMPWARE.MKMPnnn.F3	Panels
14	COMPWARE.OKMPnnn.F1	JCLIN for Japanese Language
15	COMPWARE.OKMPnnn.F2	Object Modules for Japanese Language
16	COMPWARE.MLMSnnn.F1	JCLIN
17	COMPWARE.MLMSnnn.F2	LMS load library
18	COMPWARE.MLMSnnn.F3	LMS panels, skeletons, messages and CLIST
[1] NLCXnnn is the FMID for Japanese support for CSS. OKMPnnn is the FMID for Japanese support for Base Services/HCI.		

ECC Installation Considerations

This section describes considerations for installing and customizing ECC that applies to LMS, CSS, and/or Base Services/HCI.

Link Lists

If you place an ECC component load library in the link list, it must be placed ahead of all Compuware product libraries in the link list. Refer to your Compuware product documentation to determine whether the load library or libraries for your component(s) should be placed in the link list.

PDS/E Support

ECC components support Partitioned Data Set/Extended (PDS/E) libraries. LMS, CSS, and/or Base Services/HCI can be installed in PDS/E object libraries. Input source members for the CSS preprocessor or compiled listings for the CSS postprocessor may reside in a data PDS/E library. The following are supported for PDS/E libraries:

- PDS/E object libraries are supported with DFSMS 1.1 and above.
- PDS/E data libraries are supported with DFSMS 1.0 and above.

LMS Installation Considerations

This section describes considerations for installing and customizing LMS.

APF Authorization

Following IBM system integrity guidelines, Compuware requires that your LMS load library be APF-authorized. You must provide an APF-authorized load library into which you place the modules for LMSINIT since all load modules for LMSINIT must reside in an APF-authorized load library.

Execution Sequence

In order to ensure availability of Compuware products with LMS-controlled access, the LMS runtime environment must be established before any of those products are initialized. For this reason, Compuware strongly recommends that you institute an MVS start-up procedure based on the sample proc provided. It should run automatically as part of IPL and IML processing prior to any procs utilized by other Compuware products. It may be necessary to consult your site's MVS system programmer.

The License Management System Functions

The License Management System has three basic functions.

License Management Administration

You will create and maintain your License File using the License Administration Utility (LAU) installed with your License Management Software. Your License File is used as the source from which Compuware products will validate access for your site. When you receive License Certificates for Compuware products, your organization's License Administrator must use the LAU to **import** the License Certificates into the License File. Additional features of the LAU assist in the maintenance of your License Files. In addition to browsing the License Certificates in your License File, you may run a number of administrative reports from the LAU. These reports will reveal the contents of your License File, from a number of perspectives, allowing you to determine the state of the License Certificate for any particular Compuware product release.

Depending on the requirements of your organization, you may have more than one License File. Each License File can be centrally administered from the License Administration Utility.

Establishing the Runtime LMS Environment

You will use the License File(s) created and maintained by the License Administration Utility as input to a program that establishes your Compuware LMS runtime environment. This program is named LMSINIT. LMSINIT is the License Management System program that reads the License File[1] and constructs the License Cache and License Management System subsystem against which Compuware product runtime license access requests are later made.

Important: You must re-establish your runtime License Management System environment at each IPL, and whenever you have an update to your License File, the update must be made available to your MVS or OS/390 systems. At IPL startup, LMSINIT must be completed before starting up any Compuware product.

Validating Product Access Requests During Product Use

Your access to your Compuware products will be validated when you use your Compuware products. The Compuware product will make a request of the License Management System to determine if your site has a valid License Certificate for the product release. The requests are made by the product at various times during its execution depending on the product's needs. The Compuware product will access the LMS subsystem established, request License File information and act upon the information. If the License Certificate information is valid, users will proceed with their use of the product without disruption. Any abnormal License Management event detected will be reported to the product's user and may optionally be reported by e-mail to your organization's License Administrator. Optionally, these events may also be recorded in SMF Logging.

1. LMSINIT can read more than one License File if you are an organization that administers Compuware product access for more than one Compuware customer.

CSS Installation Considerations

This section describes considerations for installing Compuware Shared Services.

APF Authorization

Following IBM system integrity guidelines, Compuware recommends that your CSS load library NOT be APF-authorized. Depending on your MVS system installation options, libraries placed in the link list may be APF-authorized automatically.

Note: If you are using Abend-AID XLS and Abend-AID is APF-authorized, then CSS should also be APF-authorized. If Abend-AID is not APF-authorized, then CSS should not be APF-authorized.

CICS Startup JCL

The CSS library must be placed ahead of all Compuware product libraries in the DFHRPL concatenation of the CICS startup JCL for the following products:

- XPEDITER/CICS (all supported releases)
- CICS Abend-AID/FX (all supported releases)

CSS TSO Logon PROCs

If the CSS load library is not in the link list and dynamic library allocation is not used, the CSS load library must be placed before all Compuware product load libraries in the STEPLIB or ISPLLIB concatenations in the TSO logon PROC. The CSS message and panel libraries must be placed before all Compuware product message and panel libraries in the ISPMLIB and ISPPLIB concatenations in the TSO logon PROC.

Beginning with CSS 6.0, the CSS load, message, and panel libraries can be dynamically allocated for use with Compuware/VF. See SLCXCNTL members CWVFCLSE, CWVFCLSJ, CWVFREXE, and CWVFREXJ.

Sharing CSS DDIO Files With Multiple CPUs

If you are sharing CSS DDIO files among multiple CPUs, the following qnames must be changed from LOCAL to GLOBAL enqueues.

- ABENDAID
- ABENDDAM
- ABENDSMF
- CWCATALG

Note: These qnames are needed for **all** products. Please refer to the individual Compuware product documents with which CSS was distributed for other product requirements.

For IBM's GRS (Global Resource Serialization), these qnames must be added to the inclusion RNL. This RNL is defined in the GRSRNLxx member of SYS1.PARMLIB for MVS/XA and later. For pre-MVS/XA systems, these qnames are maintained in the ISGSIRNL entry point of ISGGRNLO. Please consult the IBM product documentation for further details.

For resource serialization packages other than GRS (for example MIM), consult the product documentation for changing LOCAL to GLOBAL enqueues.

CSS Customer Modification Facility

The CSS Customer Modification Facility retrofits your site's CSS restricted zap(s) to the SMP/E environment. If you are migrating to the current CSS release from a CSS release prior to CSS 7.4, and have a restricted zap(s) applied to your system, you must use the CSS Customer Modification Facility. For more information on using this facility, refer to the "Using the Customer Modification Facility" appendix in the *Compuware Shared Services User/Reference Guide*.

CSS Language Processor (LP)

The language processor captures information about a compiler listing and stores it in a DDIO file. The majority of this information is gathered from the compiler listing. Optionally, SYSIN and SYSLIB are also examined.

The language processor can be run using two methods:

- preprocessor
- postprocessor

You may use the preprocessor or postprocessor (whichever is best for your site as needed.)

This section discusses the benefits of the preprocessor and the postprocessor and describes when to use them. See the appropriate language processor chapters in the Compuware Shared Services User/Reference Guide for additional information on determining when to use the pre- or postprocessor.

Preprocessor

Some information about a program module is not always available from the compiler listing. The preprocessor, in addition to gathering information from the compiler listing, gathers information from SYSIN and SYSLIB. The preprocessor provides additional functionality for XPEDITER/CICS and XPEDITER/TSO, and is required for PL/I programs that use the %NOPRINT statement.

When setting up the language processors, Compuware recommends that you use the preprocessor instead of the postprocessor. The postprocessor should only be used in situations where compiled source listings have been stored.

Preprocessor Steps

1. Determines the proper compiler options required to process the listing file.
2. Automatically invokes the compiler or assembler to compile or assemble your source program.
3. Writes the listing to the DDIO file. An enhanced listing for COBOL programs can be produced, if desired. PL/I and Assembler listings are written to SYSPRINT. C listings are written to SYSCPRT.

Preprocessor Benefits

- **Better handling of compiler errors.** By using the preprocessor, you can avoid the potential problem of processing a listing with compiler errors. The preprocessor, in conjunction with the CONDDDIO parameter, internally checks the return code from the compiler and doesn't write a DDIO listing that contains errors. For example, if you set the CONDDDIO parameter to 8, the preprocessor will write the compiler listing to the DDIO file unless the compiler return code exceeds 8. The preprocessor can process the compiler errors more effectively than the postprocessor. Compuware recommends that you use the preprocessor rather than the postprocessor.

- **Capturing suppressed source code.** When any of the following parameters are used, sections of source code can be suppressed from the compiler listing:

For COBOL	COPY SUPPRESS
For PL/I	%NOPRINT
For Assembler	PRINT OFF or PRINT NOGEN
For C	NOSHOWINC and NOEXPMAC

  The preprocessor captures this information from SYSIN and SYSLIB, or by forcing on more revealing compiler options, then altering the listing to emulate the options you requested.

- **Automated compiler options.** The postprocessor requires that certain compiler options be specified in order to process all needed sections of the compiler listing. The preprocessor can automatically pass the required options to the compiler.
- **Simplified JCL.** While the postprocessor requires that the user add a step after the compile step, the preprocessor requires only that minor modifications be made to your existing compile step.

Files Dynamically Allocated

The following files are dynamically allocated by the Compuware preprocessor. The attributes can be overridden by specifying a DD in the JCL.

Table 2-5. Files Dynamically Allocated by the Preprocessor

DDname	Attributes
CWPWRKn (0-6)	BLKSIZE=19000 SPACE(TRK,(100,80)) UNIT=SYSDA
SORTWK01	BLKSIZE=19000 SPACE(TRK,(100,80)) UNIT=SYSDA
TEMPLIN	BLKSIZE=3200 SPACE(TRK,(200,100)) UNIT=SYSDA
CWPERRM	SYSOUT=*
SYSPRINT	SYSOUT=*
SYSOUT	SYSOUT=*
CWPPRTI (PL/I)	BLKSIZE=19000 SPACE(TRK,(100,200)) UNIT=SYSDA
CWPPRTI (all other languages)	BLKSIZE=16093 SPACE(TRK,(100,200)) UNIT=SYSDA DISP=MOD
CWPPDS2	SPACE (TRK,(100,80,80)) UNIT=SYSDA

CWPPRTO is dynamically allocated as a work file (for the preprocessor). SYSPRINT is where the listing goes.

Postprocessor

The postprocessor is executed as a single step — can be a separate job. It reads in the listing created from the compiler. Information is gathered from the source listing, XREF, data maps, and object code sections of the listing. This information is stored in a DDIO file member and is used by various Compuware products.

While the preprocessor is the preferred method of loading the compiled listing to the DDIO, there are situations that necessitate the use of the postprocessor as a viable alternative. This is commonly found in production environments where the listings may be archived. In order for the listings to populate the DDIO, they must be compiled using the required options specified in the appropriate language processor chapters in the Compuware Shared Services User/Reference Guide. Note that these are usually the default options at many shops.

If the programs contain suppressed source code, certain additional control statements must be added to the language processor. Please view the Language Processor chapters in

the Compuware Shared Services User/Reference Guide for the programming language desired for more details.

If the programs are compiled with OFFSET as well as OPT(imize), certain additional control statements must be added to the language processor. Please view the Language Processor chapters in the Compuware Shared Services User/Reference Guide for the programming language desired for more details.

Postprocessor Benefits

- Exact match of compiled output listings with the executable load module. This is especially important in production environments where time is critical.
- Eliminates the risk of copybook or source code changes prior to recompiling the code since the listing reflects the code that is being executed.
- Significantly less time is needed to process source listings as opposed to recompiling the source code.

Language Processor DD Statements

The following table lists the DD statements for the Assembler, COBOL, PL/I, and C language processors. Not every DD Name applies to every language processor.

Table 2-6. Language Processor DD Statements.

DDname	Purpose
CWPWRK (0-6)	LP work files
CWPPRMO	LP parameter input dataset
CWPDDIO	Target source listing DDIO file
CWPPRTI	Compiler listing input to postprocessor
CWPPRTO	Compiler listing output from postprocessor
CWPLOAD	Option LOAD or OBJECT module input to postprocessor
CWPDECK	Option DECK module input to postprocessor
CWPERRM	LP Error Messages

Note: CWPLOAD and CWPDECK correspond to the SYSLIN and SYSPUNCH compiler outputs. If you specify compiler option OBJECT, you should use CWPLOAD in your postprocessor JCL. If you specify compiler option DECK, you should use CWPDECK in your postprocessor JCL. Do not feed the SYSLIN output into CWPDECK or the SYSPUNCH output into CWPLOAD. The updated OBJECT contained in CWPLOAD and CWPDECK should be used in the program's linkedit step. The preprocessor does not use CWPLOAD and CWPDECK.

Writing a Listing to a DDIO File

Several options exist to place listings in a DDIO file. Each option requires that some changes be made to your JCL. The type of changes needed depend on whether you are using the preprocessor or the postprocessor, and the type of debugging session. Refer to "CSS Sample Library Members" for a list of the sample JCL that can be used for the various options.

Testing and Debugging Programs

The actual method used for handling testing and debugging programs depends on whether you are preprocessing or postprocessing:

- **Preprocessing** — Run the preprocessor and link your program.

- **Postprocessing** — Compile your program, run the postprocessor, and then link your program.

With either of these methods, the language processor places a listing in the DDIO file whenever a program is compiled.

Debugging Production Programs

If the original compiler listing is available in machine-readable format, you can retrieve the listing and use it as input to the postprocessor in order to reprocess the listing and recreate it in the DDIO file. The postprocessor can process listings stored by most products.

Note: If the listing you use as input to the postprocessor contains any other information (such as translator or link edit output), in addition to the compiler information, you may not obtain the desired results.

CSS Sample Library Members

The following list describes the members of the CSS sample JCL library that you may work with during installation (depending on your installation options). These members are distributed on the CSS distribution tape and updated via SMP/E. The members will be located in the following files after the Apply and Accept steps are complete.

- COMPWARE.MLCX*nnn*.SLCXCNTL (target library)
- COMPWARE.MLCX*nnn*.ALCXCNTL (distribution library)

where *nnn* indicates the CSS release number.

To print the CSS sample library members, submit the JCL contained in library member CXPRINT after the tape has been unloaded.

AACONVRT Sample JCL to convert existing Abend-AID XLS report DDIO to shared directory report database

CWASSECD Security Exit program DSECT.

CWASSECU Sample Security Exit program.

CWCMTRHE Sample customized horizontal translation table for mixed-case English.

CWCMTRHT Sample customized horizontal translation table for uppercase English.

CWCMTRHU Sample customized horizontal translation table for mixed-case English with Euro character.

CWCMTRVE Sample customized vertical translation table for mixed-case English.

CWCMTRVT Sample customized vertical translation table for uppercase English.

CWCMTRVU Sample customized vertical translation table for mixed-case English with Euro character.

Note: The following five members are specific for Abend-AID. They allow browse access to diagnostic reports and source listings directly through ROSCOE or print access via a job submitted from ROSCOE to run in batch. Refer to the members for additional information.

CWROSBAT ROSCOE sample members.

CWROSBRW ROSCOE sample members.

CWROSCOE ROSCOE sample members.

CWROSDIR ROSCOE sample members.

CWROSPDF ROSCOE sample members.

CWUTCLSE Sample CLIST to invoke CSS online utilities with English messages and panels.

CWUTCLSJ Sample CLIST to invoke CSS online utilities with Japanese messages and panels.

CWUTREXE Sample REXX EXEC to invoke CSS online utilities with English messages and panels.

CWUTREXJ Sample REXX EXEC to invoke CSS online utilities with Japanese messages and panels.

CWVFCLPT CLIST to invoke the language processor debugging aid.

Note: The debugging aid is a diagnostic tool to be used only under the supervision of Compuware Technical Support in the event of a problem related to the language processor.

CWVFCLSE Sample CLIST to invoke Compuware/VF using English messages and panels.

CWVFCLSJ Sample CLIST to invoke Compuware/VF using Japanese messages and panels.

CWVFDB2 Sample CLIST to activate the File-AID DB2 interface to the Compuware Viewing Facility.

CWVFREXE Sample REXX EXEC to invoke Compuware/VF using English messages and panels.

CWVFREXJ Sample REXX EXEC to invoke Compuware/VF using Japanese messages and panels.

CWVFRXPT Sample REXX EXEC to invoke the Compuware language processor debugging aid.

Note: The debugging aid is a diagnostic tool to be used only under the supervision of Compuware Technical Support.

CXAADIRX Abend-AID XLS CWAASDUT DIRX sample JCL.

CXAAEXPO Abend-AID XLS CWAASDUT EXPORT command sample JCL.

CXAAIMPO Abend-AID XLS CWAASDUT IMPORT command sample JCL.

CXAAMOVE Abend-AID XLS CWAASDUT MOVE command sample JCL

CXAARPT Sample CWDDSUTL control statement for formatting report files.

CXALDDAA Sample VSAM parameters for allocating Abend-AID report files.

CXALDDSL Sample VSAM parameters for allocating source listing files.

CXALLBAA JCL for creating sequential Abend-AID XLS database files.

CXALLBLP JCL for creating sequential source listing database files.

CXALLBSD JCL for creating CICS Abend-AID/FX sequential transaction databases.

CXALLDAA JCL for creating Abend-AID XLS shared directories.

CXALLDLP JCL for creating Source Listing Shared Directories.

CXALLDS JCL to allocate a sequential DDIO file.

CXALLMC JCL for creating CICS Abend-AID/FX shared directories.

CXALLVAA JCL for creating VSAM Abend-AID XLS database files.

CXALLVLP JCL for creating VSAM Source Listing Database files.

CXALLVS JCL to allocate a VSAM DDIO file.

CXALLVSD JCL for creating CICS Abend-AID/FX VSAM transaction databases.

CXASM JCL for running the Assembler language postprocessor.

CXASMPRE JCL for running the Assembler language preprocessor.

CXC Sample JCL for running the C postprocessor.

CXCAPTUR Sample JCL for creating a tape from the output of the CSS Problem Documentation Utility.

CXCFGEXT Sample JCL to extract a source file from the configuration module in a CSS load library.

CXCFGINI Sample source configuration file.

CXCFGSET Sample JCL to build a configuration module in a CSS load library, from a source configuration file you create.

CXCIREG Sample JCL to register the Contact Information dataset.

CXCIVSAM Sample JCL to create the Contact Information dataset.

CXCOBPRE JCL for running the COBOL language preprocessor.

CXCOB1 JCL for the step to be added after the compile step in your current COBOL compile and link edit JCL. This is used in order to process the compiler listing through the Compuware language postprocessor.

CXCOB2 JCL to process COBOL compiler listings stored in machine-readable format.

CXCOB99 Sample JCL for running the COBOL postprocessor.

CXCPRE Sample JCL for running the C preprocessor.

CXC2 Sample JCL to postprocess C compiler listings stored in machine-readable format.

CXDDSUTL Sample JCL for running the CWDDSUTL batch file utility.

CXDDUNLP Sample JCL to execute the CWDDUNLP source extraction utility.

CXEXPORT Sample JCL for running the CWDDSUTL EXPORT command.

CXFLAG	Sample JCL for the utility FLAG command.
CXFMTDS	JCL for formatting a DDIO file.
CXFMTLST	Sample CWDDSUTL control statement for formatting a source listing file.
CXFXDIRX	CICS Abend-AID/FX CWFXSDUT DIRX sample JCL.
CXFXEXPO	CICS Abend-AID/FX CWFXSDUT EXPORT command sample JCL.
CXFXEXTL	Sample JCL to export a CICS Abend-AID/FX transaction **and** a source listing to a single tape.
CXFXIMPO	CICS Abend-AID/FX CWFXSDUT IMPORT command sample JCL.
CXFXMOVE	CICS Abend-AID/FX CWFXSDUT MOVE command sample JCL.
CXIMPORT	Sample JCL for running the CWDDSUTL IMPORT command.
CXJCLSEC	JCL to assemble and link edit the Security Exit program.
CXJCLTRT	JCL to assemble and link edit the custom translation tables.
CXLPASM	Sample Assembler language processor options.
CXLPC	Sample C language processor options.
CXLPCOBB	Sample COBOL language processor options (batch programs).
CXLPCOBC	Sample COBOL language processor options (CICS programs).
CXLPDIRX	LP source CWDDLPUT DIRX sample JCL.
CXLPEXPO	LP source CWDDLPUT EXPORT command sample JCL.
CXLPIMPO	LP source CWDDLPUT IMPORT command sample JCL.
CXLPMOVE	LP source CWDDLPUT MOVE command sample JCL.
CXLPPLI	Sample PL/I language processor options.
CXLPTFS	Sample JCL to list the PTFs and APARs on a CSS load library.
CXMODMAP	Sample JCL member to map the CSECTs in a load module.
CXOTFPRE	Sample CLIST stub for preprocessing exit for OTF
CXOTFPST	Sample CLIST stub for postprocessing exit for OTF
CXPDDSU	Sample JCL to print a listing or report from a DDIO file.
CXPLI	JCL for running the PL/I language postprocessor.
CXPLIPRE	JCL for running the PL/I language preprocessor.
CXPLI2	JCL to postprocess PL/I compiler listings stored in machine-readable format.
CXPRCTL	IEBPTPCH control statements for job CXPRINT.
CXPRINT	JCL to print the entire CSS installation library.

CXRELS	PTF/APAR tracking module. **Do not modify this member.**
CXR00003	Sample Tutorial main menu.
CXSPRE01	Sample Skeleton for on-the-fly preprocessing.
CXSPST01	Sample Skeleton for on-the-fly postprocessing.
CXTUTOR	Alternate Sample Tutorial main panel.
DDIODAYS	REXX utility to delete DDIO file members by age.
DDIODAYJ	Sample JCL for running the DDIODAYS REXX utility. This utility deletes DDIO file members by age.
FDXTRN01	File-AID DB2 Interface CLIST.
P@CPRE01	Sample panel for preprocessing exit for on the fly.
P@CPST01	Sample panel for postprocessing exit for on the fly.
P@HPRE01	Sample Help panel for preprocessing exit for on-the-fly processing.
P@HPST01	Sample Help panel for postprocessing exit for on-the-fly processing.

DVS Installation Considerations

This section describes considerations for installing Distributed Viewing Support (which requires that Base Services/HCI be installed).

APF Authorization

The Base Services/HCI component library (used by Distributed Viewing Support, COMPWARE.MKMPnnn.SKMPAUTH, must be APF-authorized. This library is used in the Base Services/HCI server region only.

TSO Logon PROCs

If you are utilizing Distributed Viewing Support, you should update the CWVFCLSE, CWVFCLSJ, CWVFREXE, and CWVFREXJ members to add the Base Services/HCI Viewing Support library to the ISPLLIB concatenations (COMPWARE.MKMPnnn.SKMPLOAD). It may also be added to your TSO logon PROC or added to the link list.

Chapter 3.
Installing Enterprise Common Components

IMPORTANT:

If the ECC components exist on your system, you can use the enclosed tape as you would a product maintenance tape to update the components with the latest changes. However, any ECC components which have newer release numbers MUST be installed.

CAUTION:

To prevent possible conflicts, Compuware strongly recommends that all Compuware products and components distributed via SMP/E share a single Global zone reserved solely for Compuware products. Target and distribution zones should not be shared with other products or different releases of the same product.

This chapter explains the steps required to install the primary portions of Compuware's Enterprise Common Components (ECC), including:

- License Management System (LMS)
- Compuware Shared Services (CSS)
- Host Communications Interface (Base Services/HCI) — a component used by Abend-AID XLS Distributed Viewing Support (DVS)

The procedures in this chapter unload the ECC tape and install the component(s) into datasets. Although the full ECC installation procedure is discussed in this chapter, you may only need to perform certain portions depending on which product(s) are used at your site. You should already have determined in the previous chapter which components you need to install.

Except where otherwise noted, the return code from all jobs should be zero.

The ECC installation process utilizes System Modification Program Extended (SMP/E). For more information about this type of installation, and other installation considerations, please refer to **Chapter 2, "Preparing for ECC Installation"**. For more information about SMP/E, consult the *IBM SMP/E Reference* or the *IBM SMP/E User's Guide*.

Note: If you have already installed components from the ECC tape and you receive an additional ECC tape, that tape may be used as a maintenance tape. If you are installing a maintenance tape, run job member $RSTABS to obtain current PTF abstract information. Approved PTFs are normally added to the maintenance tape on a monthly basis. If you receive more than one tape dated within a given month, the maintenance on these tapes may be identical. See **Chapter 4, "Applying Maintenance"** for complete maintenance installation instructions.

Step 1. Unload the ECC Installation Sample Library

The ECC installation sample library contains JCL used to complete the ECC installation process. This library is contained on the second file of the ECC tape. After unloading the library, you can review member $$INDEX for a description of each of the supplied members.

1. Type the JCL shown in Figure 3-1, and modify it as follows:
   a. Replace the *tttttt* value for **VOL=SER=** with the installation tape volume serial number.
   b. Replace the *vvvvvv* value for **VOL=SER=** with a valid DASD volume serial number. If a DASD volser is not required at your site, omit the DASD specification.
   c. The **DSN=COMPWARE.KCW*nnn*.INSTALL** value should indicate the name of ***your*** SMP/E installation sample library. Update the DSN as necessary to follow your site's naming conventions.
2. Submit the JCL to unload the initial install library.

Figure 3-1. JCL to Unload SMP/E Installation Sample Library

```
//ECC1A    JOB..your job card here...
//UNLOAD   EXEC PGM=IEBCOPY
//SYSUT3   DD UNIT=SYSDA,SPACE=(TRK,5)
//SYSUT4   DD UNIT=SYSDA,SPACE=(TRK,5)
//SYSPRINT DD SYSOUT=*
//TAPE     DD DSN=COMPWARE.LKCW001.F1,
//            UNIT=CART,                          <==VERIFY
//            DISP=(SHR,KEEP),
//            LABEL=(2,SL),
//            VOL=SER=tttttt                      <==VERIFY
//INSTLIB  DD DSN=COMPWARE.KCWnnn.INSTALL,        <==VERIFY
//            UNIT=SYSDA,
//            VOL=SER=vvvvvv,                     <==VERIFY
//            SPACE=(TRK,(45,15,20)),
//            DCB=(BLKSIZE=3120,LRECL=80,RECFM=FB),
//            DISP=(,CATLG)
//SYSIN    DD *
  COPY  INDD=TAPE,OUTDD=INSTLIB
/*
```

Step 2. Select Installation Components

The ECC Installation JCL Customization Facility uses panels to gather your installation information. It then uses that information to build the jobs needed to perform the SMP/E installation.

Note: The Installation JCL Customization Facility requires a minimum of ISPF/PDF version 3.5. REXX is also required.

For information on resolving error messages from the installation facility, refer to the *Enterprise Common Components Messages and Codes* guide.

Begin the process of installing LMS, CSS, and/or Base Services/HCI using the Installation JCL Customization Facility as follows:

1. Invoke the installation facility by entering the TSO EXECUTE command with the dataset name of the installation library followed by ECCINST. For example:

   ```
   TSO EX 'COMPWARE.KCWnnn.INSTALL(ECCINST)'
   ```

   where COMPWARE.KCWnnn.INSTALL is the name of your installation sample library as you entered it in the JCL in the previous step.

   The entry panel for the Installation JCL Customization Facility (Figure 3-2) will be displayed.

Figure 3-2. Selecting Components to Install

```
Compuware Enterprise Common Components Release 1.5.0
                 SMP/E Installation JCL Customization Facility

Command ==>

Place an "S" by one or more of the following components to select that
component for Installation JCL Customization, and press Enter.
To deselect a component, delete the "S"

    _   Compuware Shared Services  7.9

    _   Base Services/HCI          4.1

    _   License Management System  2.0




                      Press PF1 for help,  END to exit.
```

2. Enter an **S** to the left of each component you want to install. Remember the following:

   - If you only require LMS, select only LMS on this panel.
   - If you only require CSS, select only CSS on this panel.
   - If you use Abend-AID XLS 9.0 and more current and plan to use Distributed Viewing Support, you must install both the CSS and Base Services/HCI components.

   Note: If you are installing more than one component, Compuware recommends installing them simultaneously.

To indicate your choices, the word **SELECTED** will be displayed next to each component chosen. Figure 3-3 shows the confirmation text that would be displayed if CSS, Base Services/HCI, LMS, and AAP were all selected for installation.

Figure 3-3. Confirming Installation Selections

```
Compuware Enterprise Common Components Release 1.5.0
                SMP/E Installation JCL Customization Facility

Command ==>

Place an "S" by one or more of the following components to select that
component for Installation JCL Customization.
To deselect a component, delete the "S"

    S   Compuware Shared Services  7.9          SELECTED
    S   Base Services/HCI          4.1          SELECTED
    S   License Management System  2.0          SELECTED

Place an "S" by the following to invoke the Abend-AID product selection
screen for installation/customization processing.

    S   Abend-AID Products                      SELECTED

Press Enter to begin JCL Customization for the selected component(s).
Select all desired components before continuing.

                     Press PF1 for help,  END to exit.
```

3. To modify a selection, add or delete the **S** to the left of the component.
4. When you are satisfied with your selections, press Enter again to start the JCL customization process.

Step 3. Specify ECC Installation Environment Information

The panel shown in Figure 3-4 is displayed when you pressed Enter to confirm your component selections in the previous step.

Figure 3-4. Installation Environment Panel

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI (DVS)
             License Management

Please enter the following parameters.

Do you want to enter SMS parameters?        NO
DASD Unit:                                  SYSDA
DASD Volser (optional):
SMP/E Work Unit:                            VIO
Installation Tape Volser:                   CN0000
Tape unit:                                  3490
Does your installation use CA-TMS?          NO
GLOBAL SMP/E dataset High-level Qualifier:  COMPWARE.GLOBAL


  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

This panel prompts you to enter information about the installation environment. Notice that the names of the components selected for installation are listed below the Command prompt. Those names will remain there during the rest of this procedure.

1. Specify the installation tape volume serial number, device, and tape unit for the ECC tape as follows:

   SMS Parameters

   If you specify YES in the SMS Parameters field, you will be prompted to enter SMS (Storage Management Subsystem) parameters on the next panel. Depending on your site's SMS installation options, default SMS parameters may be supplied automatically. In this case, you may not need to specify SMS parameters to the Installation JCL Customization Facility. If you want to supply SMS parameters, specify YES in this field. Otherwise specify NO. The default value is NO.

   DASD Unit

   Enter a valid DASD unit name. The value of this field will be used in the UNIT= parameter of the DD statements that allocate non-VSAM libraries and datasets. The default value is SYSDA.

   DASD Volser

   This field is optional. If a volume is specified, SMP/E will use it as the target for library and dataset allocation.

   SMP/E Work Unit

   Enter the name of a DASD unit to use for allocating SMP/E work files. The default value is VIO.

   Installation Tape Volser

   Enter the serial number of the ECC tape.

Tape Unit

Enter the value for the UNIT= parameter of the tape input DD statements. This is a site-dependent parameter. CART, T3490, TAPE, etc. are examples of typical unit parameters. Enter an appropriate value based on your site's standards. The default value is 3490.

CA-TMS

Enter YES or NO in the CA-TMS field. CA-TMS sometimes requires a special parameter (EXPDT=98000) on the DD statement to process labeled tapes created at another site. Specify YES to add this value to the generated JCL members. If your site does not use CA-TMS or does not require this parameter, specify NO. The default value is NO.

Global SMP/E dataset High-Level Qualifier

Enter a value to be used for generating the names of the SMP/E system datasets. These datasets include SMPLOG, SMPPTS, SMPTLIB, and the Global Consolidated Software Inventory (CSI).

Note: Due to limits on the length of the DSPREFIX parameter (used to build the SMPTLIB names for SMP/E RELFILES), this field cannot be longer than 26 characters.

2. When you have completed the fields as required, press Enter to continue.

Step 4. Specify SMS Parameters

If, in the previous step, you did not specify that you wanted to enter SMS parameters, you can skip this step.

If you specified, on the Installation Environment Panel (Figure 3-4 on page 3-5), that you wanted to enter SMS parameters, the panel shown in Figure 3-5 is displayed when you press Enter.

Figure 3-5. Specifying SMS Parameters

```
                        Enterprise Common Components
                   SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI (DVS)
             License Management

Enter any or all of the following SMS parameters.

Data Class:        dataclas

Storage Class:     storclas

Management Class:  mgmtclas




  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

1. Specify the data class, storage class, and/or management class parameters for the SMS operating environment as follows:

   Data Class

   Specify the eight-character data class name for SMS.

   Storage Class

   Specify the eight-character storage class name for SMS.

   Management Class

   Specify the eight-character management class name for SMS.

2. When you complete the fields as required, press Enter to continue.

Step 5. Specify Global CSI Dataset Name

The panel shown in Figure 3-6 is displayed when you pressed Enter in the previous step.

Figure 3-6. Verifying Global CSI Dataset Name

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI (DVS)
             License Management

The following dataset name will be used for the Global CSI:
COMPWARE.GLOBAL.CSI

If you do not want to use this CSI, type in a new CSI name.
Erase the field to set it to the default.

The dataset currently does not exist.




  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

Note: The DSN for a new global CSI cannot exceed 38 characters. If the global CSI is predefined outside of the Customization Facility, it can consist of up to 44 characters.

1. Verify that the global CSI dataset name is correct. Change the name or specify a new name if necessary.
2. Press Enter.

   The Installation JCL Customization Facility determines whether the dataset currently exists. One of the following will occur:

If:	**Then:**
The dataset exists but has not been set up for use with Compuware products.	A jobstream is generated to add the appropriate parameters to the CSI. This job is named $*xx*I0GBA, where *xx* specifies the components you are installing: • **LM** for LMS only • **CS** for CSS only • **DV** if installing only the Base Services/HCI component used by Abend-AID XLS Distributed Viewing Support • **EC** for two or more of the above components Review the resulting jobstream and compare the generated parameters to any existing SMP/E parameters.
The dataset exists and is already set up for use with Compuware products.	Job $*xx*I0GBA is not created.
The dataset does not exist.	Jobstream $xxI0GBA is generated to build the dataset and initialize it with the required Compuware SMP/E parameters.

Step 6. Specify Assembler and Jobcard Information

The panel shown in Figure 3-7 is displayed when you pressed Enter in the previous step. It prompts you for assembler and job card information.

Figure 3-7. Entering Assembler and Job Card Information

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI
             License Management

Default Assembler: ASMA90

Enter Job Card Information:
  //$$$$$$$$ JOB ('ACCT'),'USER NAME',
  //         CLASS=?,MSGCLASS=?,NOTIFY=PROGRAMMER
  //*        OPTIONAL JES PARAMETER CARD COULD GO HERE
  //*
  //*

NOTE: The class selected MUST allow enough CPU time for long running jobs.
      Jobs $xxI2APL and $xxI3ACC can easily exceed 5 CPU minutes each.


  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

1. Specify the default assembler and job card information for the Installation JCL Customization Facility as follows:

   Default Assembler

   Some Compuware product installations require the assembly of one or more modules. Enter the program name of an assembler that resides in the linklist. The default assembler in the linklist for OS/390 1.2 and more current is ASMA90. The default prior to OS/390 1.2 is IEV90.

   Job Card Information

   Update the supplied model job card to follow your site standards. You may enter up to five lines of job information.

   By default, the installation facility will set the *$$$$$$$$* value equal to the member name of each job created. You also have the option of specifying the job name using the format *xxxxxxx$* or *xxxxxxxx*, where the *x*'s are whatever name you choose. If you set it to *xxxxxxx$*, the installation facility will set the job name for each job to *xxxxxxx* and append a number to it. A letter will be appended for jobs 10 and above. If you set it to *xxxxxxxx*, the installation facility will use the specified name for all jobs created.

   Note: The job class specified **must** allow enough CPU time for long-running jobs.

2. After you have supplied the required information, press Enter to continue.

Step 7. Specify SMP/E High-level Qualifiers

The panel shown in Figure 3-8 is displayed when you pressed Enter in the previous step. It prompts you for the SMP/E high-level qualifiers of the components previously selected for installation.

Figure 3-8. Specifying SMP/E High-Level Qualifiers

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI

             License Management

Enter the following information:
  Compuware Shared Services
  SMP/E Dataset High-level Qualifier: COMPWARE.MLCXnnn

  Base Services/HCI
  SMP/E Dataset High-level Qualifier: COMPWARE.MKMPnnn

  License Management
  SMP/E Dataset High-level Qualifier: COMPWARE.MLMSnnn




  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

Note: If a high-level qualifier is longer than 28 characters, the installation facility is not able to generate a default dataset name for the target and distribution zones. You can either change the high-level qualifier entered, or manually modify the zone dataset names to valid lengths.

1. Specify the LMS, CSS, and/or Base Services/HCI high-level qualifiers as applicable for your installation as follows:

   CSS SMP/E Dataset High-Level Qualifier

   Enter a value to be used in generating names of CSS-specific datasets, including target and distribution zone CSIs, and their respective non-SMP/E target and distribution libraries. The value entered cannot exceed 35 characters.

   Base Services/HCI SMP/E Dataset High-Level Qualifier

   Enter a value to be used in generating names of Base Services/HCI-specific datasets, including target and distribution zone CSIs, and their respective non-SMP/E target and distribution libraries. The value entered cannot exceed 35 characters.

   LMS SMP/E Dataset High-Level Qualifier

   Enter a value to be used in generating names of LMS-specific datasets, including target and distribution zone CSIs, and their respective non-SMP/E target and distribution libraries. The value entered cannot exceed 35 characters.

2. After you supply the appropriate high-level qualifiers, press Enter. You will be prompted for the LMS, CSS, and/or Base Services/HCI target and distribution zones, depending on which components you are installing.
3. Continue with the appropriate step(s) below, skipping any that do not apply.

Step 8. Specify CSS Target and Distribution Zones

If you are not installing CSS, you may skip this step.

If you are installing the CSS component, you will be prompted to specify the CSS target and distribution zones as shown in Figure 3-9.

Figure 3-9. Specifying CSS Target and Distribution Zones

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI
             License Management

Enter the following information for Compuware Shared Services

Distribution Zone Name:            LCXnnnD
DZone CSI Dataset Name:            COMPWARE.MLCXnnn.DZONE.CSI
DZone CSI currently does not exist


Target Zone Name:                  LCXnnnT
TZone CSI Dataset Name:            COMPWARE.MLCXnnn.TZONE.CSI
TZone CSI currently does not exist

Japanese Language Support?         NO

Warning: Dzone and/or Tzone name differ from default for CSS release 7.8
Zone CSI exists, has Zone Name already been defined?     NO

  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

1. Specify the CSS target and distribution zone information as follows:

   Distribution Zone Name

   Specifies the name of the SMP/E distribution zone used by CSS. The name you specify must be unique within the global zone. The default is LCXnnnD.

   DZone CSI Dataset Name

   Specifies the dataset name of the SMP/E distribution zone used by CSS. The default is COMPWARE.MLCXnnn.DZONE.CSI.

   Distribution CSI Currently exists/does not exist

   Indicates whether the CSS distribution zone CSI dataset exists.

   Target Zone Name

   Specifies the name of the SMP/E target zone used by CSS. The name you specify must be unique within the global zone. The default is LCXnnnT.

   Target Zone CSI Dataset Name

   Specifies the dataset name of the SMP/E target zone used by CSS. The default is COMPWARE.MLCXnnn.TZONE.CSI.

   Target CSI Currently exists/does not exist

   Indicates whether the CSS target zone CSI dataset exists.

   Japanese Language Support

   CSS can allocate message and panel libraries to display Japanese language characters rather than English. Specify YES to allocate and install the Japanese language message and panel libraries. Specify NO to bypass allocating these libraries.

Warning: Dzone and/or Tzone name differ from default for CSS release 7.8

You will get this warning message only if the distribution and/or target zone names entered are different from the default.

Zone CSI exists, has Zone Name already been defined?

You will get this Zone message if the CSI(s) for both the Target and Distribution Zones already exist. See Figure 3-10 for an explanation.

Figure 3-10. Installing into Pre-existing Target/Distribution Zones

```
                        Enterprise Common Components
              SMP/E Installation JCL Customization Facility - Tutorial

CAUTION: If the CSI(s) for both the Target and Distribution Zones already
exist we will ask you whether you are installing into an existing set of
Target and Distribution Zones.  If you reply NO or you allow it to default,
but you are installing into an existing set of Target and Distribution Zones,
there is a possibility of destroying data in or related to the existing Zones.

Compuware strongly suggests you install components into new Zones.  If you
wish to install into the same or pre-existing CSIs then specify the CSI
name(s) in the Zone CSI Datasetname fields.
```

Note: You will only get the above screen if the Zone CSI exists message appears on Figure 3-9 and you reply NO.

2. After you supply the appropriate information, press Enter. You may be prompted for the LMS and/or Base Services/HCI target and distribution zones, depending on which components you are installing.
3. Continue with the appropriate step(s) below, skipping any that do not apply.

Step 9. Specify Base Services/HCI Target and Distribution Zones

If you are not using Abend-AID XLS Distributed Viewing Support, you may skip this step.

If you are installing the Base Services/HCI component, you will be prompted to specify the target and distribution zones as shown in Figure 3-11.

Figure 3-11. Specifying Base Services/HCI Target and Distribution Zones

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI
             License Management

Enter the following information for Base Services/HCI

Distribution Zone Name:            KMPnnnD
DZone CSI Dataset Name:            COMPWARE.MKMPnnn.DZONE.CSI
DZone CSI currently does not exist


Target Zone Name:                  KMPnnnT
TZone CSI Dataset Name:            COMPWARE.MKMPnnn.TZONE.CSI
TZone CSI currently does not exist

Japanese Language Support?         NO

Warning: Dzone and/or Tzone name differ from default for HCI release 4.1
Zone CSI exists, has Zone Name already been defined?     NO

  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

1. Specify Base Services/HCI target and distribution zone information as follows:

   Distribution Zone Name

   Specifies the name of the SMP/E distribution zone used by Base Services/HCI. The name you specify must be unique within the global zone. The default is KMPnnnD.

   DZone CSI Dataset Name

   Specifies the dataset name of the SMP/E distribution zone used by Base Services/HCI. The default is COMPWARE.MKMPnnn.DZONE.CSI.

   Distribution CSI Currently exists/does not exist

   Indicates whether the Base Services/HCI distribution zone CSI dataset exists.

   Target Zone Name

   Specifies the name of the SMP/E target zone used by Base Services/HCI. The name you specify must be unique within the global zone. The default is KMPnnnT.

   Target Zone CSI Dataset Name

   Specifies the dataset name of the SMP/E target zone used by Base Services/HCI. The default is COMPWARE.MKMPnnn.TZONE.CSI.

   Target CSI Currently exists/does not exist

   Indicates whether the target zone CSI dataset exists.

   Japanese Language Support

Abend-AID XLS Distributed Viewing Support can display Japanese language characters rather than English. Specify YES to install Japanese language support. Specify NO to bypass installing this support.

Warning: Dzone and/or Tzone name differ from default for Base Services/HCI release 4.1

You will get this warning message only if the distribution and/or target zone names entered are different from the default.

Zone CSI exists, has Zone Name already been defined?

You will get this Zone message if the CSI(s) for both the Target and Distribution Zones already exist. See Figure 3-12 for an explanation.

Figure 3-12. Installing into Pre-existing Target/Distribution Zones

```
                        Enterprise Common Components
            SMP/E Installation JCL Customization Facility - Tutorial

 CAUTION: If the CSI(s) for both the Target and Distribution Zones already
 exist we will ask you whether you are installing into an existing set of
 Target and Distribution Zones.  If you reply NO or you allow it to default,
 but you are installing into an existing set of Target and Distribution Zones,
 there is a possibility of destroying data in or related to the existing Zones.

 Compuware strongly suggests you install components into new Zones.  If you
 wish to install into the same or pre-existing CSIs then specify the CSI
 name(s) in the Zone CSI Datasetname fields.
```

Note: You will only get the above screen if the Zone CSI exists message appears on Figure 3-11 and you reply NO.

2. After you supply the appropriate information, press Enter. You may be prompted for the LMS target and distribution zones, depending on which components you are installing.
3. Continue with the step below, or skip it if it does not apply.

Step 10. Specify LMS Target and Distribution Zones

If you are not installing LMS, you may skip this step.

If you are installing LMS, you will be prompted to specify the LMS target and distribution zones as shown in Figure 3-13.

Figure 3-13. Specifying the LMS Target and Distribution Zones

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI
             License Management

Enter the following information for License Management

Distribution Zone Name:            LMSnnnD
DZone CSI Dataset Name:            COMPWARE.MLMSnnn.DZONE.CSI
DZone CSI currently does not exist


Target Zone Name:                  LMSnnnT
TZone CSI Dataset Name:            COMPWARE.MLMSnnn.TZONE.CSI
TZone CSI currently does not exist


Warning: Dzone and/or Tzone name differ from default for LMS release 2.0
Zone CSI exists, has Zone Name already been defined?     NO



  Press Enter to continue, PF1 for help, or END to return to previous panel.
```

1. Specify the LMS target and distribution zone information as follows:

   Distribution Zone Name

   Specifies the name of the SMP/E distribution zone used by LMS. The name you specify must be unique within the global zone. The default is LMSnnnD.

   DZone CSI Dataset Name

   Specifies the dataset name of the SMP/E distribution zone used by LMS. The default is COMPWARE.MLMSnnn.DZONE.CSI.

   Distribution CSI Currently exists/does not exist

   Indicates whether the distribution zone CSI dataset exists.

   Target Zone Name

   Specifies the name of the SMP/E target zone used by LMS. The name you specify must be unique within the global zone. The default is LMSnnnT.

   Target Zone CSI Dataset Name

   Specifies the dataset name of the SMP/E target zone used by LMS. The default is COMPWARE.MLMSnnn.TZONE.CSI.

   Target CSI Currently exists/does not exist

   Indicates whether the target zone CSI dataset exists.

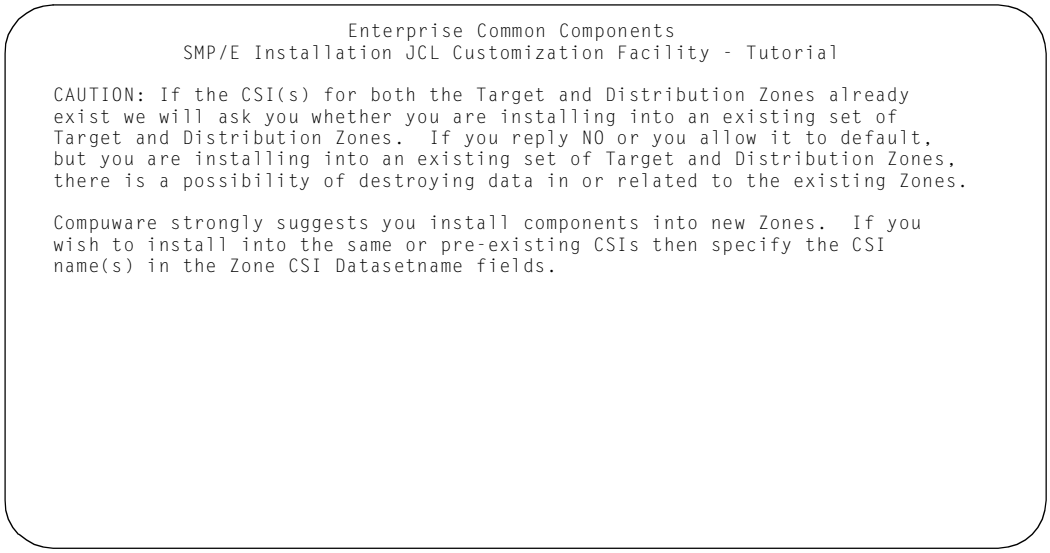
   Warning: Dzone and/or Tzone name differ from default for LMS release 2.0

   You will get this warning message only if the distribution and/or target zone names entered are different from the default.

   Zone CSI exists, has Zone Name already been defined?

You will get this Zone message if the CSI(s) for both the Target and Distribution Zones already exist. See Figure 3-14 for an explanation.

Figure 3-14. Installing into Pre-existing Target/Distribution Zones



Note: You will only get the above screen if the Zone CSI exists message appears on Figure 3-13 and you reply **NO**.

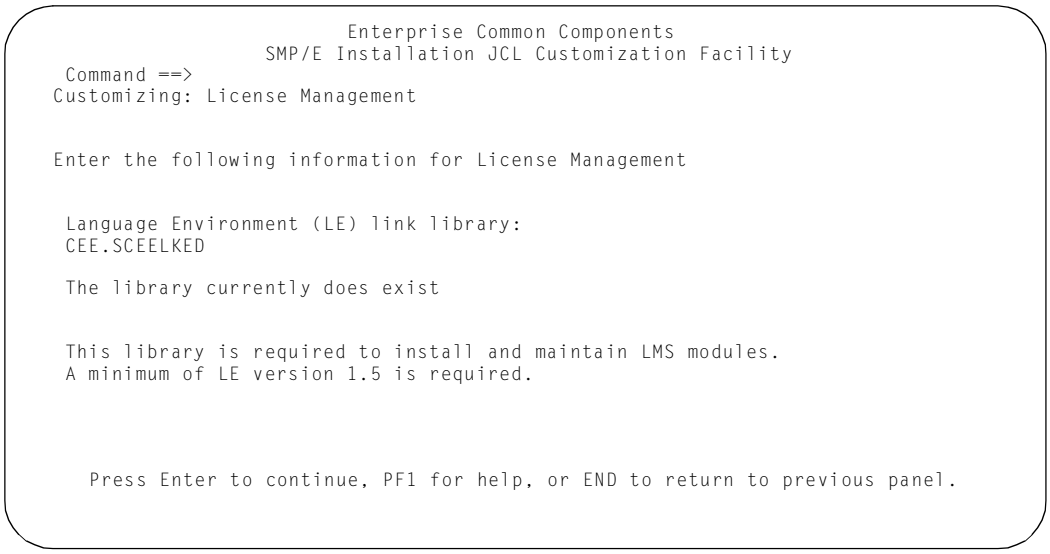
2. After you supply the appropriate information, press Enter.

Step 11. Specify Language Environment Link Library for LMS

If you are not installing LMS, you may skip this step.

If you are installing LMS, you will be prompted to specify the IBM Language Environment (LE) link library, as shown in Figure 3-15.

Figure 3-15. Specifying the Language Environment Link Library for LMS



1. Specify the IBM Language Environment link library as follows:

   Language Environment (LE) link library

   > Beginning with LMS Release 2.0, you must specify your site's LE link library for the proper installation and maintenance of LMS.

   After you enter the LE link library dataset name, this screen displays whether the library exists.

2. After you enter the library name and verify its existence, press Enter.

Step 12. Verify Installation Information

When you pressed Enter in the previous step, the first of a series of confirmation panels was displayed. These panels verify all of the information you have entered up to this point.

- To accept the information on each confirmation panel, press Enter.
- To correct a particular parameter, press END to return to the associated entry panel.

After you accept the information on each of the confirmation panels, the installation process will continue.

Step 13. Process the JCL

The panel shown in Figure 3-16 is displayed when you pressed Enter the final time in the previous step.

Figure 3-16. Confirming JCL Processing

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility

                             Processing Verification Screen
Command ==>
Customizing: Compuware Shared Services       Base Services/HCI
             License Management

Press Enter to begin customizing the JCL.
```

1. Press Enter to start the processing that builds customized JCL members. Press END to cancel the installation.

   When you press Enter, the progress of the JCL customization process is displayed on the screen. After the installation JCL has been created, a panel is displayed informing you that a parameter report has been generated (Figure 3-17). This report details the JCL created and lists the parameters and execution information used during installation.

Figure 3-17. Report Member Detailing JCL Processing

```
                        Enterprise Common Components
                  SMP/E Installation JCL Customization Facility

The following PDS member contains a report of all parameters used during
customization and a log of all JCL members created:


COMPWARE.KCWnnn.INSTALL(xxREPORT)


                 Press Enter to browse the report, END to exit.
```

2. If you want to browse the report at this time, press Enter. You can also examine the report later by accessing the applicable installation library report member shown on this panel:
   - LMREPORT (LMS only)
   - CSREPORT (CSS only)
   - DVREPORT (Base Services/HCI only)
   - ECREPORT (Two or more components)
3. The processing of the Installation JCL Customization Facility is now finished. Press END to exit.

Jobs Created by the Installation JCL Customization Facility

All of the jobs created during installation customization are described below. Note that in addition to component-specific jobs, a series of jobs used during the maintenance cycle are also created. See “Maintenance Jobs” on page 3-21 for more information.

Jobs Common to All Components

The jobs in this section are customized for use when you install an ECC component. The *xx* that appears in the job names is replaced with a two-letter code during installation. Which two-letter code is used depends on which component(s) you are installing:

- **LM** (LMS only)
- **CS** (CSS only)
- **DV** (Base Services/HCI only)
- **EC** (Two or more components)

$xxI0GBA Allocates and/or initializes the Compuware Global CSI and Global SMP/E datasets.

Note: This job is not built if the Global CSI already exists and you indicate it is already set up for Compuware use.

$xxI1PRA Allocates the LMS, CSS, and/or Base Services/HCI libraries and SMP/E datasets.

$xxI1PRB Initializes the LMS, CSS, and/or Base Services/HCI SMP/E target and distribution zone CSIs. It then adds the required LMS, CSS, and/or Base Services/HCI entries to the Global Zone CSI.

$xxI2RCV Receives the LMS, CSS, and/or Base Services/HCI base code and maintenance.

$xxI3APL Applies the LMS, CSS, and/or Base Services/HCI base code and maintenance.

$xxI4ACC Accepts the LMS, CSS, and/or Base Services/HCI base code and maintenance.

$PRTABS Optional job to print abstracts of all PTFs in the maintenance file.

$RSTABS JCL to unload current PTF abstracts from an installation or maintenance tape.

LMS Installation Jobs

The following jobs are customized for installing LMS only.

$LMPABS Optional job that lists current LMS PTF abstracts.

$LMLPTFS Lists all PTFs applied to LMS target and distribution zones. This optional job is for reporting purposes only.

CSS Installation Jobs

The following jobs are customized for installing CSS only.

$CXPABS Optional job that lists current CSS PTF abstracts.

$CXLPTFS Lists all PTFs applied to the CSS target and distribution zones. This optional job is for reporting purposes only.

Base Services/HCI Installation Jobs

The following jobs are customized for installing Base Services/HCI only.

$DVPABS Optional job that lists current Base Services/HCI PTF abstracts.

$DVLPTFS Lists all PTFs applied to the Base Services/HCI target and distribution zones. This optional job is for reporting purposes only.

Maintenance Jobs

The following jobs may also be built by the installation facility but are used for applying maintenance PTFs. These jobs are used when a maintenance tape is applied. For more information on applying maintenance, refer to **Chapter 4, "Applying Maintenance"**.

LMS Jobs

- $LMM0RCV
- $LMM1RCD
- $LMM2APL
- $LMM3ACC

CSS Jobs

- $CSM0RCV
- $CSM1RCD
- $CSM2APL
- $CSM3ACC

Base Services/HCI Jobs

- $DVM0RCV
- $DVM1RCD
- $DVM2APL
- $DVM3ACC

Step 14. Allocate Compuware Global CSI and SMP/E Datasets

Note: If your site previously installed a Compuware product using SMP/E, skip this step.

This step allocates and initializes the Compuware Global CSI for use with SMP/E. It also allocates the SMPLOG and SMPPTS datasets. For more information on these datasets, refer to Table 2-3 on page 2-5.

The installation facility has created a job to allocate the required datasets based on which components were chosen for installation at your site:

$LMI0GBA (LMS only)

$CSI0GBA (CSS only)

$DVI0GBA (Base Services/HCI only)

$ECI0GBA (Two or more components)

1. Review the appropriate job for any necessary changes.
2. Submit the job to start the allocation process.

Step 15. Allocate Libraries and SMP/E Datasets

This step allocates target and distribution zone CSIs as well as product-specific SMP/E datasets.

The installation facility has created a job to allocate the required libraries and datasets based on which components were chosen for installation at your site:

$LMI1PRA (LMS only)

$CSI1PRA (CSS only)

$DVI1PRA (Base Services/HCI only)

$ECI1PRA (Two or more components)

Note: The space allocations used are based on 3390 DASD. If you use 3380 DASD, you may decide to change the BLKSIZE for the load library and APF authorized load library to 6356 for a better blocking factor.

1. Review the appropriate job for any necessary changes.
2. Submit the job to start the allocation process.

Step 16. Initialize the Target and Distribution Zone CSIs

This step initializes the LMS, CSS, and/or Base Services/HCI target and distribution zone CSIs.

The installation facility has created a job to initialize the target and distribution zone CSIs based on which components were chosen for installation at your site:

$LMI1PRB (LMS only)

$CSI1PRB (CSS only)

$DVI1PRB (Base Services/HCI only)

$ECI1PRB (Two or more components)

1. Review the appropriate job for any necessary changes.

   Note: Before you submit this job, make sure the job submitted in the previous step has completed.

2. Submit the job to start the initialization process.

   Note: This job may return RC=4 if either of the following messages are issued:

   - GIM56501W The aaaaaa subentry was added rather than replaced because it did not exist.
   - GIM27701W aaaaaa entry bbbbbbb was added rather than replaced because it did not exist.

   Otherwise, the return code should be 0.

Step 17. Receive the Base Code and Maintenance PTFs

This step receives the LMS, CSS, and/or Base Services/HCI base code and maintenance PTFs from the SMP/E tape into the SMPPTS and SMPTLIBs.

The installation facility has created a job to receive the base code and maintenance PTFs and to accept the base code depending on which components were chosen for installation at your site:

$LMI2RCV (LMS only)

$CSI2RCV (CSS only)

$DVI2RCV (Base Services/HCI only)

$ECI2RCV (Two or more components)

1. Review the appropriate job for any necessary changes.
2. Submit the job to start the initialization process.
3. Review any HOLDDATA listed by the job (if any) and take appropriate actions.

CAUTION:
Be sure to run $LMI2RCV, $ECI2RCV, $CSI2RCV, or $DVI2RCV in a job class suitable for long-running batch jobs.

Step 18. Apply the Base Code and Maintenance PTFs to the Target Libraries

This step applies the LMS, CSS, and/or Base Services/HCI base code and maintenance PTFs to the target zone(s).

The installation facility has created a job to apply the base code to the target zone depending on which components were chosen for installation at your site:

$LMI3APL (LMS only)

$CSI3APL (CSS only)

$DVI3APL (Base Services/HCI only)

$ECI3APL (Two or more components)

1. Review the appropriate job for any necessary changes.
2. Submit the job to Apply the base code and maintenance PTFs.

CAUTION:
Be sure to run $LMI3APL, $ECI3APL, $CSI3APL, or $DVI3APL in a job class suitable for long-running batch jobs.

Step 19. Accept the Base Code and Maintenance PTFs to the Distribution Libraries

This step accepts the LMS, CSS, and/or Base Services/HCI base code and maintenance PTFs into the distribution libraries.

The installation facility has created a job to accept the base code into the distribution libraries depending on which components were chosen for installation at your site:

$LMI4ACC (LMS only)

$CSI4ACC (CSS only)

$DVI4ACC (Base Services/HCI only)

$ECI4ACC (Two or more components)

1. Review the appropriate job for any necessary changes.
2. Submit the job to Accept the base code and maintenance PTFs.

CAUTION:
Be sure to run $LMI4ACC, $ECI4ACC, $CSI4ACC, or $DVI4ACC in a job class suitable for long-running batch jobs.

Note: This job may return RC=4 if either of the following messages are issued:

- GIM24701W SMP/E could not obtain link-edit parameters for load module xxxxxxxx for sysmod xxxxxxxx. Defaults were used.
- GIM23903W Link-edit processing for SYSMOD xxxxxxxx was successful for module mmmmmmmm in LMOD llllllll in the ALCXLOAD library. The return code was 04.

These messages can be ignored.

Step 20. Continue Installation and Customization

Now that the Component Base Code and Maintenance have been installed, there may be further customization tasks that need to be addressed. Depending on which components you have installed, continue with the following chapters as necessary:

- **Chapter 5, "LMS Customization"**
- **Chapter 6, "CSS Customization"**

Chapter 4.
Applying Maintenance

IMPORTANT

If the ECC components exist on your system, you can use the enclosed tape as you would a product maintenance tape to update the components with the latest changes. However, any ECC components which have newer release numbers MUST be installed. If the latest updates for ECC components are already installed, you do not need to reinstall them.

This chapter describes the procedure for applying ECC maintenance.

- Because all Compuware mainframe products utilize the License Management System, you should perform all applicable procedures in this chapter related to LMS. These are typically identified by a reference to LMS following the job name.
- If you use a Compuware product that utilizes Compuware Shared Services, you should also perform all applicable procedures in this chapter related to CSS. These are typically identified by a reference to CSS following the job name.
- Base Services/HCI is required only for Abend-AID XLS Distributed Viewing Support. You should only apply Base Services/HCI maintenance if you use Abend-AID XLS Distributed Viewing Support. These jobs are identified by a reference to Base Services/HCI following the job name.

Note: CICS Abend-AID/FX also distributes a version of Base Services/HCI that may or may not be the same release level as used by Abend-AID XLS Distributed Viewing Support. If you are a CICS Abend-AID/FX user, the Base Services/HCI maintenance on this tape may already have been applied from the CICS Abend-AID/FX maintenance tape.

Your ECC installation tape includes all maintenance for the included components available as of the day the tape was created. You may request a separate, cumulative maintenance tape from Compuware Technical Support that contains the latest preventive service. You may also obtain individual PTFs from Compuware to address specific problems you report in priority 1 situations. Maintenance is installed using SMP/E.

If you ran the Installation JCL Customization Facility in the initial installation (see “Step 3. Specify ECC Installation Environment Information” on page 3-5), the following jobs have already been customized:

- $LMM0RCV (LMS), $CSM0RCV (CSS), or $DVM0RCV (Base Services/HCI) performs an SMP/E receive of the maintenance tape.

  ($LMM1RCD, $CSM1RCD, or $DVM1RCD performs an SMP/E receive of maintenance from disk.)
- $LMM2APL (LMS), $CSM2APL (CSS), or $DVM2APL (Base Services/HCI) performs an SMP/E apply to target libraries.
- $LMM3ACC performs an SMP/E accept to LMS distribution libraries.
- $CSM3ACC performs an SMP/E accept to CSS distribution libraries.
- $DVM3ACC performs an SMP/E accept to Base Services/HCI distribution libraries.

The following steps are recommended for the maintenance cycle:

1. Accept any PTFs from a previous maintenance run that have not been accepted yet.

2. Receive and Apply the new maintenance PTFs.
3. Test the new PTFs for any problems that may arise due to site dependencies, then move them into production and accept the new PTFs. You could also wait until the next maintenance run before doing the accept.

   Note: The ECC maintenance tape includes a separate file that contains updated PTF abstracts ($LMSABS, $LCXABS, $KD1ABS). See "Step 1b. Unload PTF Abstracts from Maintenance Tape" on page 4-3 to execute job $RSTABS, which unloads these members into the SMP/E installation sample library.

CAUTION:

Products using CSS should be shut down while maintenance is being applied to their CSS libraries and restarted after the maintenance process is complete. This applies specifically to CICS Abend-AID/FX and XPEDITER/CICS users.

For more information about using SMP/E, consult the *SMP/E Reference* or the *SMP/E User's Guide* for the release of SMP/E you are using.

Step 1a. Unload the ECC SMP/E Installation Sample Library

This step is applicable only if you no longer have the SMP/E installation sample library available. If you still have the library from the initial installation, you may skip to Step 1b.

The SMP/E installation sample library contains JCL samples for completing the installation process. The JCL shown in Figure 4-1 will unload this library. It is unloaded as part of the installation process, see "Step 1. Unload the ECC Installation Sample Library" on page 3-2. This library is contained on the second file of the maintenance tape.

Note: The ttttt should be replaced by the maintenance tape volume serial number and vvvvvv should be replaced by a valid DASD volume serial number. For KCWnnn, nnn should be replaced by the ECC release number.

After you unload the SMP/E installation sample library, execute the Installation JCL Customization Facility to build the following maintenance jobs:

- $LMM0RCV (LMS), $CSM0RCV (CSS), or $DVM0RCV (Base Services/HCI)
- $LMM1RCD (LMS), $CSM1RCD (CSS), or $DVM1RCD (Base Services/HCI)
- $LMM2APL (LMS), $CSM2APL (CSS), or $DVM2APL (Base Services/HCI)
- $LMM3ACC (LMS), $CSM3ACC (CSS), or $DVM3ACC (Base Services/HCI).

Figure 4-1. JCL To Unload ECC Maintenance Tape

```
//ECC1A    JOB..your job card here...
//UNLOAD   EXEC PGM=IEBCOPY
//SYSUT3   DD UNIT=SYSDA,SPACE=(TRK,5)
//SYSUT4   DD UNIT=SYSDA,SPACE=(TRK,5)
//SYSPRINT DD SYSOUT=*
//TAPE     DD DSN=COMPWARE.LKCW001.F1,
//            UNIT=CART,                          <==VERIFY
//            DISP=(SHR,KEEP),
//            LABEL=(2,SL),
//            VOL=SER=tttttt                      <==VERIFY
//INSTLIB  DD DSN=COMPWARE.KCWnnn.INSTALL,        <==VERIFY
//            UNIT=SYSDA,
//            VOL=SER=vvvvvv,                     <==VERIFY
//            SPACE=(TRK,(45,15,20)),
//            DCB=(BLKSIZE=3120,LRECL=80,RECFM=FB),
//            DISP=(,CATLG)
//SYSIN    DD *
  COPY  INDD=TAPE,OUTDD=INSTLIB
/*
```

Step 1b. Unload PTF Abstracts from Maintenance Tape

If you executed Step 1a, you may skip to Step 2.

This step restores the most current version of the PTF abstracts to the SMP/E installation sample library.

Note: If the $RSTABS member is **NOT** in the sample library, execute Step 1a to unload the sample library to a temporary PDS. Then copy members $LMSABS, $RSTABS, $LCXABS, $KD1ABS, and $KMPABS to your SMP/E installation sample library and continue with Step 2.

Update member $RSTABS (as shown in Figure 4-2) with the maintenance tape volser number and, if necessary, the dataset name of the SMP/E Installation Sample library. Submit the job to restore the PTF abstracts into the installation library.

Figure 4-2. $RSTABS JCL

```
//jobname JOB (jobcard information)
//*
//* CHECK THAT THE OUT1 DSN BELOW POINTS TO THE INSTALLATION PDS
//* REPLACE 'TTTTTT' WITH THE MAINTENANCE TAPE VOLSER
//* REPLACE 'NNN' WITH THE ECC RELEASE NUMBER
//*
//RSTRABS     EXEC PGM=IEBCOPY  UNLOAD PTF ABSTRACTS
//SYSPRINT    DD SYSOUT=*
//SYSUDUMP    DD SYOUT=*
//INI         DD DSN=COMPWARE.LKCW001.F5,
//               UNIT=CART,
//               EXPDT=98000,
//               VOL=(,RETAIN,SER=TTTTTT),
//               LABEL=(6,SL),
//               DISP=(,KEEP)
//OUT1       DD  DISP=SHR,DSN=COMPWARE.KCWnnn.INSTALL
//SYSIN      DD  *
  COPY  OUTDD=OUT1,INDD=((IN1,R))
/*
```

Step 2. Receive ECC Maintenance

Use SMP/E Installation Sample Library member **$LMM0RCV** (LMS), **$CSM0RCV** (CSS), or **$DVM0RCV** (Base Services/HCI) to receive any maintenance from the maintenance tape. Review this JCL and change *ttttt* to match the maintenance tape volume serial number, if necessary. $LMM1RCD (LMS), $CSM1RCD (CSS), or $DVM1RCD (Base Services/HCI) will receive maintenance from disk datasets.

After you review and modify the JCL, submit the job to start the receive process.

Note: If the job completes with RC=4, review the output for the following message:

```
GIM24801W NO SYSMODS SATISFIED THE OPERANDS SPECIFIED ON THE
RECEIVE COMMAND
```

If this message appears, no preventive service is applicable for the specified FMID(s).

Step 3. Review HOLDDATA

At times, installing a maintenance tape requires additional actions that must be performed after a PTF(s) is applied. In these cases, the PTF is identified as an exception with:

```
++HOLD
```

and the HOLDDATA contains information on the additional actions you must perform to completely implement the change.

It is very important that you perform any HOLDDATA action. Failure to do so can cause unpredictable results, including incorrect data and abends. It is possible that the maintenance tape does not have any hold actions. In this case, the HOLDDATA dataset will be empty.

Note: Jobs $LMM0RCV (LMS), $CSM0RCV (CSS), or $DVM0RCV (Base Services/HCI) issue a command to display any HOLDDATA on the maintenance tape.

Step 4. Apply ECC Maintenance

Use the SMP/E installation sample library member **$LMM2APL** (LMS), **$CSM2APL** (CSS), or **$DVM2APL** (Base Services/HCI) to apply any maintenance to the target libraries.

Notes:

1. Run the job in a job class suitable for long-running batch jobs.
2. You may be required to have SMP/E restore a USERMOD prior to applying a PTF to a module. SMP/E messages inform you when this action is necessary. After the PTF is applied, you must then reapply the USERMOD.
3. It is advisable to run an APPLY CHECK before applying any maintenance.

The following parameter must be added to the APPLY command of $CSM2APL or $LMM2APL to bypass any HOLDDATA: BYPASS(HOLDUSER(ACTION)).

After you review and modify the JCL, submit the job to start the apply process.

Step 5. Accept ECC Maintenance

CAUTION:
Compuware strongly recommends that you do not accept any PTFs until they have been thoroughly tested at your site.

After a function or maintenance (PTFs or APARs) has gone through the ACCEPT step, that becomes your current release-base level. At any time prior to the ACCEPT step, you may issue a RESTORE command to reset your target libraries to the way they were as of the last ACCEPT command. The RESTORE command removes maintenance, or even entire functions, while the ACCEPT command effectively sets a new base level of the release to which further maintenance will be applied. For more information about using the RESTORE command, refer to your site's IBM SMP/E documentation.

Use the SMP/E installation sample library member **$LMM3ACC** (LMS), **$CSM3ACC** (CSS), or $DVM3ACC (Base Services/HCI) to have SMP/E accept any maintenance to the distribution libraries. The following parameter must be added to the ACCEPT command of $LMM3ACC or $CSM3ACC to bypass any HOLDDATA: BYPASS(HOLDUSER(ACTION)). Job $DVM3ACC may require parameter BYPASS(HOLDSYSTEM) to bypass any HOLDDATA. Submit the job to start the accept process.

Notes:

1. Run the job in a class suitable for long-running batch jobs.
2. It is advisable to run an ACCEPT CHECK before accepting the **PTFs**.
3. Do **NOT** perform an ACCEPT on any **USERMODs**!

Step 6. Continue with ECC Customization

Note: After maintenance has been applied, you must continue with the procedures in Chapter 5, "LMS Customization".

Resolving Problems

If any error messages were returned while performing steps in this chapter, refer to the *Enterprise Common Components Messages and Codes* guide.

If problems persist, contact Compuware Technical Support.

Chapter 5.
LMS Customization

This chapter provides the basic procedures for setting up and customizing Compuware's License Management System (LMS) for use with the Compuware products licensed by your organization. It includes instructions for:

- Transferring a License Certificate to your host system
- Setting up and using the License Administration Utility (LAU) to create a License File and define system nodes
- Importing a Certificate into a License File
- Initializing the LMS runtime environment.

Preliminary Considerations

Which tasks you need to perform will depend on whether LMS has already been enabled at your site.

Running More than One License Management System

Under certain circumstances, a site may decide to run more than one instance of LMS on an MVS system. For instance, you might want both test and production LMS environments on the same MVS system. To make the production LMS environment the default LMS environment for the MVS system, designate it in the LMSINIT JCL and startup proc prepared in "Step 7. Prepare an APF Authorized LMSINIT" on page 5-13. Designate a different name for the test LMS environment in a separate LMSINIT JCL and proc. Then add a DD statement, pointing to the test LMS subsystem, to the startup code of the Compuware products you want to utilize that environment. To use different License Certificates in the test LMS environment, create a separate test LMS License File and use it as input to the LMSINIT JCL and proc for your LMS test environment. For more information on multiple LMS environments, see the chapter entitled "Creating Runtime Environment" in the *License Management System User/Reference Guide*.

Managing More than One Compuware Customer Number

A site that manages multiple Compuware customer numbers, for example a service bureau, will have a License File for each Compuware customer number. All of the relevant License Files for a system can be loaded into a single LMS environment by including the datasets as input when preparing the LMSINIT JCL and proc as described in "Step 7. Prepare an APF Authorized LMSINIT" on page 5-13, then running LMSINIT as described in "Step 8. Run LMSINIT" on page 5-20. By using the file access security set up by the service bureau for their customers, LMS provides control over which customers can use which License File. For more information, refer to the appendix entitled "Service Bureau Environment" in the *License Management System User/Reference Guide*.

Note: Any use of Compuware products by third parties must be authorized in your License Agreement.

Which Steps Do I Need to Perform?

The choice of which steps to perform depends on whether LMS has already been installed and initialized as part a previous Compuware product installation. For this reason, the procedures in this chapter are tailored for two possible scenarios:

- You need to install and set up LMS for the first time, then enable it to work with the Compuware product being installed.
- You are an established LMS user who only needs to enable it to work with the Compuware product being installed.

As shown in Table 5-1, a first-time user must perform every step. Most established LMS users can skip Steps 3, 4, and 5.

Table 5-1. LMS Steps Required

First-Time User	Step	Established User
x	"Step 1. Transfer License Certificate to Host"	x
x	"Step 2. Set up License Administration Utility (LAU)"	x
x	"Step 3. Create License File" on page 5-6	
x	"Step 4. Verify License File" on page 5-8	
x	"Step 5. Define Nodes" on page 5-9	
x	"Step 6. Import License Certificate" on page 5-11	x
x	"Step 7. Prepare an APF Authorized LMSINIT" on page 5-13	x
x	"Step 8. Run LMSINIT" on page 5-20	x
x	"Step 9. Verify Proper LMS Function" on page 5-21	x
x	"Step 10. Continue ECC Installation" on page 5-22	x

Installing and Enabling LMS

Follow the instructions below that apply to your situation.

Step 1. Transfer License Certificate to Host

LMS uses License Certificate files to configure access to Compuware products licensed by your organization. A License Certificate is a text file typically sent to your site via e-mail by Compuware's Worldwide License Management team. To be used by LMS, a License Certificate must be accessible to MVS.

1. Locate the License Certificate for the Compuware product(s) being installed. A License Certificate file can contain licensing for more than one Compuware product.
2. Allocate a target dataset on the host system for the License Certificate. Use the DCB parameters RECFM=FB, LRECL=80, and whatever BLKSIZE you prefer.
3. Transfer the License Certificate to the host using File Transfer Protocol (FTP), IND$FILE, cut and paste, or any other method desired.

   Note: If you open the License Certificate in an ISPF editor or transfer it using cut-and-paste, make sure NUMBERS is set to OFF.

4. If you have more than one License Certificate file, repeat the process as required.
5. Proceed as follows:
   - If you are a first-time user or an LMS 1.0 user, continue with "Step 2. Set up License Administration Utility (LAU)".
   - If you are an established LMS 2.0 user, go to "Step 4. Verify License File" on page 5-8.

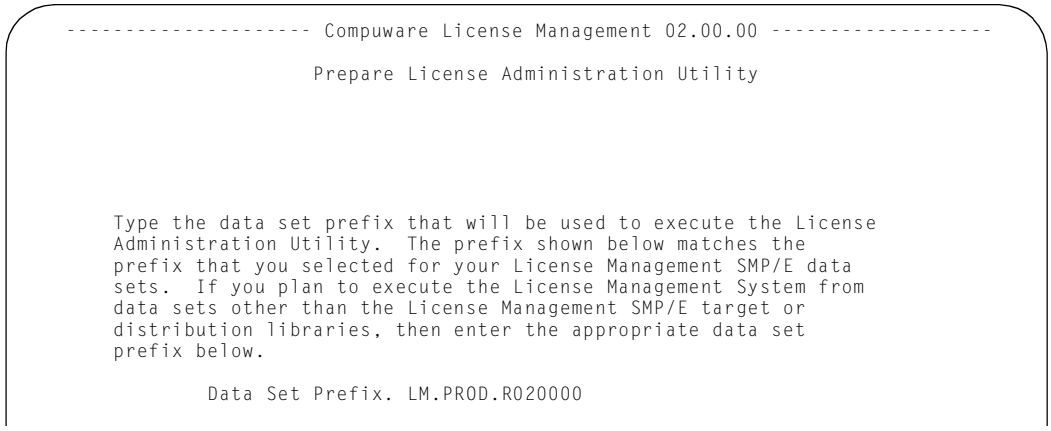
Step 2. Set up License Administration Utility (LAU)

This step creates the CLIST for starting the LAU, prepares the LAU to run, and makes it accessible to the License Administrators at your site.

1. Enter the following command to execute the LMSPREP EXEC from the SLMSCNTL library.

   ```
   TSO EX ' your SMP/E dataset prefix.SLMSCNTL(LMSPREP)'
   ```

Figure 5-1. Prepare License Administration Utility Screen



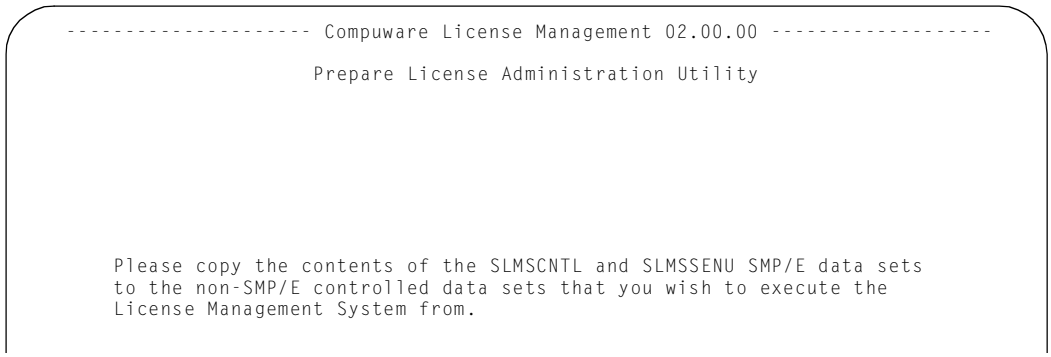
```
---------------------- Compuware License Management 02.00.00 -------------------

                      Prepare License Administration Utility




    Type the data set prefix that will be used to execute the License
    Administration Utility.  The prefix shown below matches the
    prefix that you selected for your License Management SMP/E data
    sets.  If you plan to execute the License Management System from
    data sets other than the License Management SMP/E target or
    distribution libraries, then enter the appropriate data set
    prefix below.

            Data Set Prefix. LM.PROD.RO20000
```

The following messages will notify you that you have successfully tailored the SLMSCNTL screen.

```
LMSP100I - SLMSCNTL(CWLMA) SUCCESSFULLY TAILORED
LMSP100I - SLMSSENU(LMASTPLB) SUCCESSFULLY TAILORED
LMSP100I - SLMSCNTL(CWLMQURY) SUCCESSFULLY TAILORED
LMSP100I - SLMSCNTL(CWLMDMSG) SUCCESSFULLY TAILORED
LMSP100I - SLMSCNTL(CWLMCHKP) SUCCESSFULLY TAILORED
```

2. The second screen will be displayed only when the dataset prefix field displayed in the first screen was changed to a value that did not match the dataset prefix of the SLMSCNTL library that contained LMSPREP. This is to accommodate customers who do not want to execute the LAU directly from the SMP/E target libraries for the License Management System.

Figure 5-2. Prepare License Administration Utility - second screen



```
---------------------- Compuware License Management 02.00.00 -------------------

                      Prepare License Administration Utility




    Please copy the contents of the SLMSCNTL and SLMSSENU SMP/E data sets
    to the non-SMP/E controlled data sets that you wish to execute the
    License Management System from.
```

3. Perform either of the following:
   - Add the SLMSCNTL library to your SYSPROC concatenation, or
   - Copy member CWLMA from the SLMSCNTL library into a dataset already concatenated to your SYSPROC DD.
4. Add the following line to an ISPF selection panel of your choice:

   ```
   L,'CMD(%CWLMA) NOCHECK'
   ```

   Note: Adding CWLMA to ISPF is optional, and you may execute CWLMA as a CLIST.

5. If your site has elected to secure the ISPF Command Table via an external security manager (i.e., RACF, TOPSECRET, or ACF2), then you must take appropriate steps to identify the command LMAMAIN to your external security manager.

Step 3. Create License File

If you have previously installed Compuware's License Management System software, go to "Step 4. Verify License File" on page 5-8.

In this step, you will use the LAU to create a License File.

1. Start the LAU by selecting the ISPF menu item that was added as part of "Step 2. Set up License Administration Utility (LAU)" on page 5-4. The License File Selection screen (Figure 5-3) will be displayed.

Figure 5-3. License File Selection Screen

```
 ---------------------- Compuware License Management 02.00.00 -- Row 1 to 2 of 2
Command ===>                                                   SCROLL ===> PAGE
                            License File Selection

Current Selection:
Enter New DSN  . .
                 (fully qualified without quotes)
   Delete/Define . N (Y|N)

 OR select below:-

Action     DSN                                              Added by

****************************** Bottom of data ********************************
```

2. Type the name you want to use for your License File (fully qualified without quotes) in the Enter New DSN field.
3. Type **Y** in the Delete/Define field.
4. Press Enter. The Delete/Define and Initialize License File screen (Figure 5-4) will be displayed, and the file name you specified above will be shown in the New License File field.

Figure 5-4. Delete/Define and Initialize License File Screen

```
---------------------- Compuware License Management 02.00.00 -- Row 1 to 2 of 2
Command ===>                                                   SCROLL ===> PAGE
                    Delete/Define and Initialize License File

New License File . : TSOID01.License.File.New

 Unit . . . . . . . . _________   (required for JES3 only)
Volume Serial  . . . _______     (blank for system determined volume)

Edit JCL . . . . . . Y  (Y-Yes,N-No)

     Press END Key to skip this process

Jobcard:
//Job Card information line 1
//Job Card information line 2
//Job Card information line 3
//Job Card information line 4

Select     Node      Description

****************************** Bottom of data ********************************
```

5. Type **Y** in the Edit JCL field.
6. Type the information for your jobcard in the four lines of the Jobcard field.
7. Press Enter. The Edit Delete/Define JCL screen (Figure 5-5) will be displayed.

Figure 5-5. Edit Delete/Define JCL Screen

```
 File Edit Confirm Menu Utilities Compilers Test Help
-------------------------------------------------------------------------------
EDIT       SYS98274.T095731.RA000.TSOID01.R0111344            Columns0000100080
Command ===>                                                   Scroll ===>CSR
****************************************** Top of Data ***************************
000001 //Job Card information line 1
000002 //Job Card information line 2
000003 //Job Card information line 3
000004 //Job Card information line 4
000005 //*
000006 //*********************************************************************
000007 //* DELETE/DEFINE COMPUWARE LICENSE FILE
000008 //*
000009 //* JCL GENERATED BY TSOID01 ON 2001-05-17 AT 15:22
000010 //*********************************************************************
000011 //*
000012 //LFDELDEF EXEC PGM=IDCAMS
000013 //*
000014 //SYSPRINT DD SYSOUT=*
000015 //SYSIN DD *
000016 DELETE (TSOID01.License.File.New)
000017 SET MAXCC = 0
000018 DEF CL(NAME(TSOID01.License.File.New)
000019        IXD
```

8. Make any necessary edits, then enter the END command. A Confirm Submission window will be displayed over the Delete/Define and Initialize License File screen.
9. Type **Y** in the Confirm Submission window and press Enter. The JCL to create your License File will be submitted.
10. Enter the END command. The License File Selection screen (Figure 5-3) will be displayed.
11. After your job has completed with a good return code, type **S** in the Action field next to your new License File and press Enter. Your License File name will appear in the Current Selection field.
12. Enter the END command.
    - If you are a first-time user, the main Compuware License Management screen (Figure 5-8 on page 5-9) will appear.
    - If you are an established user, the Parameter Option screen (Figure 5-9 on page 5-9) will appear. To access the main Compuware License Management screen (Figure 5-8 on page 5-9), enter the END command.

Step 4. Verify License File

In this step, you will use the LAU to verify which License File you will be using.

1. Start the LAU by selecting the ISPF menu item that was added as part of "Step 2. Set up License Administration Utility (LAU)" on page 5-4. Because the LAU has been run before and a License File already exists, the main Compuware License Management screen shown in Figure 5-8 on page 5-9 will be displayed. If the dataset name of the License file that you want to use is displayed next to the Browse option, go to "Step 5. Define Nodes" on page 5-9.
2. Type 0 in the Option field and press Enter. The Parameter Option screen (Figure 5-6) will be displayed.

Figure 5-6. Parameter Option Screen

```
 ---------------------- Compuware License Management 02.00.00 ------------------
Option ===>

 1)   Select      License File - TSOID01.LICENSE.FILE
 2)   Node        Specify System Nodes




 X) Exit          Return to previous panel









      (C) Copyright 1998, Compuware Corp. All Rights Reserved.
```

3. Type 1 in the Option field and press Enter. The License File Selection screen (Figure 5-7) will be displayed.

Figure 5-7. License File Selection Screen

```
 ---------------------- Compuware License Management 02.00.00 -- Row 1 to 2 of 2
Command ===>                                                  SCROLL ===> PAGE
                            License File Selection

Current Selection:
Enter New DSN  . .
                (fully qualified without quotes)
   Delete/Define . N (Y|N)

OR select below:-

Action     DSN                                          Added by
  _        LICENSE.FILE1                                USERID1  2000-08-07
  _        LICENSE.FILE2                                USERID1  2000-08-07
****************************** Bottom of data ********************************
```

4. Type **S** in the Action field next to the file that you want to use and press Enter. Your selected file should appear in the current selection field.

Step 5. Define Nodes

If your installation does not consist of multiple nodes in a JES network, or if you do not require the ability to route License Management work (Import, Export, or Reporting) from one JES node to another in your network, go to "Step 6. Import License Certificate" on page 5-11.

In this step, you will use the LAU to define (to LMS) the network nodes on which License Management maintenance or reporting tasks could be dispatched.

Figure 5-8. Main Compuware License Management Screen

```
 ---------------------- Compuware License Management 02.00.00 ------------------
Option ===>

 0)    Parameters Specify System Parameters
 1)    Browse     License File - TSOID01.License.File
 2)    Update     License File
 3)    IMPORT     Import License Certificate
 4)    EXPORT     Export License Certificate
 5)    Reports    Run Reports

 6)    Disaster   Enable Disaster Site
 7)    Emergency  Emergency Password

 X)    Exit License Management




        Copyright (c) 1999 Compuware Corporation. All Rights Reserved.
Unpublished rights reserved under the Copyright Laws of the United States.
          Enter HELP for Copyright/Trade Secret Notice information.
```

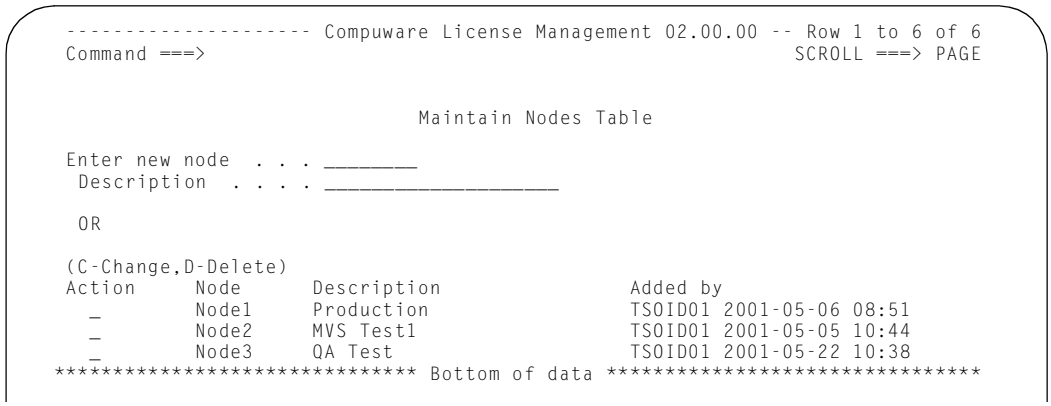
1. Type **0** in the Option field and press Enter. The Parameter Option screen (Figure 5-9) will be displayed.

Figure 5-9. Parameter Option Screen

```
 ---------------------- Compuware License Management 02.00.00 ------------------
Option ===>

 1)   Select      License File - TSOID01.LICENSE.FILE
 2)   Node        Specify System Nodes




 X) Exit          Return to previous panel




      (C) Copyright 1998, Compuware Corp. All Rights Reserved.
```

2. Type **2** in the Option field and press Enter. The Maintain Nodes Table screen (Figure 5-10) will be displayed.

Figure 5-10. Maintain Nodes Table Screen



3. In the Enter new node field on the Maintain Nodes Table (Figure 5-10), specify the name of the network node you want to add to the display list. This name identifies your installation's local JES in a network of systems or system complexes being used for network job entry (NJE) tasks. The node name can be up to eight alphanumeric characters.
4. Describe the new node briefly in the Description field. This description can be up to 20 alphanumeric characters.
5. Press Enter. The screen will be redisplayed with the new node in the selection list at the bottom of the screen. Repeat tasks step 3 through step 5 until you have defined all of the required nodes.
6. Enter the END command. The Parameter Option screen (Figure 5-9) will be displayed.
7. Enter the END command again. The main Compuware License Management screen (Figure 5-8) will be displayed.

Step 6. Import License Certificate

In this step the License Certificate for the Compuware product being installed will be imported into your site's License File.

1. On the main Compuware License Management screen (Figure 5-8 on page 5-9), type **3** in the Option field and press Enter. The first of two Import License Certificate screens (Figure 5-11) will be displayed.

Figure 5-11. First Import License Certificate Screen

```
---------------------- Compuware License Management 02.00.00 ------------
 Command ===>

                            IMPORT License Certificate

 Process DSN  . : TSOID01.License.File

 IMPORT From  . . LICENSE.CERTIF____________________
                                          (fully qualified without quotes)
 Preview Only . . N (Y|N)
```

2. Type the name of the dataset used for your License Certificate file (fully qualified without quotes) in the IMPORT From field.
3. Type **N** in the Preview Only field.

   Note: If you prefer to generate and review a report detailing the implications of the License Certificate importation, enter Y in the Preview Only field. You then need to repeat this step with Preview Only set to N.

4. Press Enter. The second Import License Certificate screen (Figure 5-12) will be displayed.

Figure 5-12. Second Import License Certificate Screen

```
---------------------- Compuware License Management 02.00.00 -- Row 1 to 5 of 6
Command ===>                                                  SCROLL ===> PAGE

                            IMPORT License Certificate

Process DSN  . : TSOID01.License.File
IMPORT From  . : LICENSE.CERTIF
Preview Only . : N


Edit JCL . . . . Y (Y-Yes,N-No)

Jobcard:
//Job Card information line 1
//Job Card information line 2
//Job Card information line 3
//Job Card information line 4

Select     Node      Description
  _        Node1     Production
  _        Node2     MVS Test1
  _        Node3     QA Test
****************************** Bottom of data ***********************
```

5. Type **Y** in the Edit JCL field.
6. Type the information for your job card in the four lines of the Jobcard field.

Note: If you are not using all four lines of the job card, comment out the unused lines.

7. Type **S** next to the node(s) on which you want the Import job to run. If no node is selected, the job is submitted without any specific routing.

   Note: If you are running on only one node, do not make a selection here.

8. Press Enter. The Edit JCL screen (Figure 5-13) will be displayed.

Figure 5-13. Edit JCL Screen

```
 File Edit Confirm Menu Utilities Compilers Test Help
--------------------------------------------------------------------------------
EDIT SYS98274.T095731.RA000.TSOID01.R0111344              Columns0000100080
Command ===>                                                 Scroll ===>CSR
********************************** Top of Data ******************************
000001 //Job Card information line 1
000002 //Job Card information line 2
000003 //Job Card information line 3
000004 //Job Card information line 4
000005//*
000006//*********************************************************************
000007//* IMPORT COMPUWARE LICENSE CERTIFICATE
000008//*
000009//* JCL GENERATED BY TSOID01 ON 2001-07-08 AT 15:32:17
000010//*********************************************************************
000011//*
000012//IMPORT EXEC PGM=LMALFIM,
000013// PARM='UPDATE'
000014//STEPLIB DD DSN=LM.DEVL.LOAD,
000015// DISP=SHR
000016//*
000017//CWLF0000 DD DSN=TSOID01.License.File,
000018// DISP=SHR
000019//CWLFIMP DD DSN=LICENSE.CERTIF,
```

9. Make any necessary edits, then enter the END command. A Confirm Submission window will be displayed over the second Import License Certificate screen.
10. Type **Y** in the Confirm Submission window and press Enter. The JCL to create your License File will be submitted.
11. Enter the END command. The main Compuware License Management screen (Figure 5-8 on page 5-9) will be displayed.
12. Type **X** in the Option field and press Enter to exit the LAU.
13. Proceed as follows:
    - If you are a first-time user, continue with "Step 7. Prepare an APF Authorized LMSINIT".
    - If you are an established user, go to "Step 8. Run LMSINIT" on page 5-20.

Step 7. Prepare an APF Authorized LMSINIT

If you are an established user, please ensure that the modules listed below are copied from your License Management SMP/E target load library into your License Management APF authorized load library and go to "Step 8. Run LMSINIT" on page 5-20.

In this step, you will ensure that the LMS modules will be loaded from an APF authorized library, and you will also create the customized proc, JCL, and parameters dataset that will be used by LMSINIT to build the LMS runtime environment.

1. In order to function correctly, the LMS load modules must reside in an MVS APF authorized library. Consult your site's MVS system programmer for details. This can be accomplished in either of the following ways:
   - APF authorize the library in which all the load modules for LMSINIT currently reside.
   - Move the following LMSINIT dataset members into an existing APF-authorized library:
     - LMACG000
     - LMADSALC
     - LMADTCV
     - LMALFAC
     - LMSAESTA
     - LMASFAC
     - LMSASSYS
     - LMSCHKPT
     - LMSCPUID
     - LMSDCTLG
     - LMSDIDCM
     - LMSDMAIN
     - LMSDSV99
     - LMSEMERP
     - LMSENMGR
     - LMSGETMS
     - LMSICRET
     - LMSIDLET
     - LMSIDUMP
     - LMSIESTA
     - LMSIFRR
     - LMSILAKC
     - LMSIMAIN
     - LMSINIT (link edited with AC=1)
     - LMSIPARM
     - LMSIPRNT
     - LMSIREAD
     - LMSISMF
     - LMSISRB
     - LMSIUPDT
     - LMSMSGEN
     - LMSMSGLT
     - LMSPCARR
     - LMSPCRTN
     - LMSRACF
     - LMSRSMGR
     - LMSSCOMM
     - LMSSDUMP
     - LMSSERVR
     - LMSSMF
     - LMSTRACE

2. Parameters used by LMSINIT are read from a SYSIN dataset. Edit the sample dataset provided in SLMSCNTL(LMINPARM) Figure 5-14).
   - Multiple operand identifiers and their associated values are allowed on a single record in the input file.
   - The operand and its associated value can start in any column and can be continued on more than one statement.
   - Columns 1 through 71 are available for use.
   - Column 72 is reserved.
   - Columns 73 through 80 can be used for sequence numbers. They are ignored by LMSINIT.
   - To continue an operand on a second statement, complete the first statement through column 71, and continue the operand on the next statement starting in column 1.
   - Only the value specified for SUBSYSTEM_ID is case-sensitive. All other operands and their values are folded to upper case by LMSINIT.
   - Only one occurrence of each of the operands is allowed. If duplicates exist, the last one encountered will be used.
   - An asterisk '*' in column 1 causes the entire statement to be a comment.
   - Line level comments are supported using the '/*' to start a comment and '*/' to end the comment. Imbedded comments are supported.

   For more information about specific parameters, refer to the chapter entitled "Creating Runtime Environment" in the *License Management System User/Reference Guide*.

   Note: The following parameters are required: FUNCTION(UPDATE), SITE(), SUBSYSTEM_ID(), CHKPT_DSNAME(). CHKPT_DSNAME() is new with release 2.0 of LMS.

Figure 5-14. Sample LMSINIT Parameters Dataset

```
*****************************************************************
**   LMINPARM - SAMPLE LMSINIT PARAMETER FILE                  **
*****************************************************************
*
*****************************************************************
**                                                             **
**    The following are REQUIRED parameters for LMSINIT        **
**    execution, preceded by a brief description of valid      **
**    entries.                                                 **
**                                                             **
**    Consult the ECC: LMS USER REFERENCE GUIDE for a          **
**    more detailed description of these LMSINIT parameters    **
*****************************************************************
*****************************************************************
**    UPDATE - Creates a new LMS Subsystem and/or
**             updates an existing subsystem
**    CREATE - Creates a new LMS Subsystem
**    DELETE - Deletes an existing LMS Subsystem
*****************************************************************
*
   FUNCTION(UPDATE)          /* CREATE, UPDATE, OR DELETE */
*
*****************************************************************
**   This number (or numbers) tell LMSINIT which site(s) to
**   load from the license file(s) into the license cache.
**   Site numbers are defined in the license certificates.
**
**   There are three different formats of the SITE operand.
**
**   1. SITE(nnn)
**   2. SITE(ddname,nnn)
**   3. SITE((ddname1,nnn),(ddname2,mmm).....,(ddnamen,ppp))
*****************************************************************
*
   SITE()                    /* 3 DIGIT SITE NUMBER       */
*
```

```
*****************************************************************
**    The value specified on the SUBSYSTEM_ID operand names
**    the OS/390 subsystem that this invocation of LMSINIT
**    is to process. This value can contain upper and lower
**    case letters and numbers, and any special characters.
**
**    It is necessary to coordinate the use of subsystem
**    identifier, such that no duplicate names exist,
**    either within the set of License Management
**    subsystems, or across all subsystems defined to OS/390.
**    You should check with your systems programmer to ensure
**    that the SUBSYSTEM_ID value you have chosen is not
**    already in use.
*****************************************************************
*
   SUBSYSTEM_ID()            /* 4 CHARACTER SUBSYSTEM ID  */
*
*****************************************************************
**    This operand defines whether this SUBSYSTEM_ID is to
**    be declared the default subsystem DEFAULT(YES),
**    DEFAULT(YES,FORCE) or is not DEFAULT(NO).
**
**    Default Subsystems require special authority to define.
**
**    See "The Default Subsystem" topic in the ECC: LMS USER
**    REFERENCE GUIDE for a discussion of how this parameter
**    affects license processing.
*****************************************************************
*
   DEFAULT()               /* "NO" "YES" OR "YES,FORCE" */
*
*****************************************************************
**
**    LMS 2.0 requires a checkpoint data set be available to
**    LMSINIT and to certain operating system exits at all
**    times. This data set is created automatically by LMINIT
**    if it does not already exist, or it is updated if it does
**    exist.  You must specify the name of this data set,
**    either using the parameters described here, or by a
**    special DD statement in the LMSINIT JCL.
**
*****************************************************************
*
*****************************************************************
*
**    CHKPT_DSNAME specifies the 1 to 44 character dsname of
**    the LMS 2.0 checkpoint data set. LMSINIT will create this
**    data set automatically, if it does not already exist.
**    NOTE: The USERID under which LMSINIT is run MUST have
**    ALTER access to the security system (RACF, ACF/2,
**    TOPSECRET) entity named by this DSNAME. The IDCAMS
**    utility is dynamically invoked to define this data set.
**    See the other (optional) checkpoint data set parameters
**    later in this member.
**
**
*****************************************************************
*
   CHKPT_DSNAME()
*
*****************************************************************
**    The following are OPTIONAL parameters for LMSINIT      **
**    execution, preceded by a brief description of valid    **
**    entries.                                               **
**                                                           **
**    Consult the ECC: LMS USER/REFERENCE GUIDE for a more   **
**    detailed description of these LMSINIT parameters.      **
*****************************************************************
*
*****************************************************************
**    The SERVICE_BUREAU operand controls the use of an      **
**    external security manager (RACF, ACF/2, or TOPSECRET)  **
**    in determining a user's access to licenses in the      **
**    cache.  This operand is used ONLY for invocations      **
**    of LMSINIT that contain multiple //CWLFxxxx DD         **
**    statements or specifies multiple SITE values.          **
*****************************************************************
*
   SERVICE_BUREAU()         /* "YES" OR "NO" */
*****************************************************************
**    The value specified on the SMF_ID operand defines the
**    SMF Record ID number that is to be used on all SMF
**    records written by the License Management System.
**    The SMF_ID must be a 3 digit number in the range of
**    128 through 255.
```

```
*************************************************************
*
   SMF_ID()                 /* 1-3 DIGIT SMF RECORD ID   */
*
*************************************************************
**    The value specified on the GTF_ID operand defines the
**    GTF Record ID number that is to be used on all GTF
**    records written by the License Management System.
**
**    Note:  You should not specify a GTF_ID unless your are
**    diagnosing a license management program error and are
**    directed by Compuware to invoke GTF recording.
*************************************************************
*
   GTF_ID()                 /* 1-5 DIGIT GTF RECORD ID   */
*
*************************************************************
**    The value specified on the LANGUAGE operand specifies
**    the language that error messages are to use.  This two
**    character operand can specify the following languages:
**
**    EN - English
**
**    If LANGUAGE is not specified,
**    then 'EN' (English) is chosen.
**
*************************************************************
*
   LANGUAGE()               /* 2 CHAR COUNTRY CODE       */
*
*************************************************************
**    LMSINIT has the ability to load an emergency license
**    cache even if the license file is unavailable or has
**    become corrupted.
**
**    The value required on the EMERGENCY operand is obtained
**    by calling the Worldwide License Management department
**    at Compuware, and by requesting an emergency password.
**    This password can also be entered using the License
**    Administration Utility to update the license file.
**
**    If, however, the license file or LAU is not available
**    you can specify the EMERGENCY password in the LMSINIT
**    parameter dataset.  Then remove all CWLFnnnn DD
**    statements from the LMSINIT execution JCL and submit
**    the JCL.
**
**    When a legitimate emergency password is present in the
**    LMSINIT sysin dataset, all Compuware products will be
**    allowed to execute on the CPUs that LMSINIT established
**    a license cache for.
**
**    The password includes its expiration date.
**    The expiration date can be from 2 to 14 days from the
**    date of issue.
*************************************************************
*
   EMERGENCY()              /* EMERGENCY PASSWORD        */
*
*************************************************************
*
**    CHKPT_VOLSER specifies the 6 character volume serial of
**    the DASD volume on which the checkpoint data set is to
**    reside.  This parameter is optional. If it does not exist,
**    your system installation defaults for VSAM data sets will
**    be used to determine the placement of this data set. You
**    must insure that any VOLSER your specify is consistent
**    the SMS class definitions that may also exist.
**
*************************************************************
*
   CHKPT_VOLSER()
*
*************************************************************
*
**    CHKPT_STORCLASS, CHKPT_DATACLASS and CHKPT_MGMTCLASS
**    specifies the names your installation has chosen to
**    describe the allocation of this VSAM checkpoint data set.
**    These parameters are optional, but if specified, will be
**    used within the IDCAMS DEFINE control statements when
**    the data set is created.
**
*************************************************************
*
   CHKPT_STORCLASS()
```

```
   CHKPT_DATACLASS()
   CHKPT_MGMTCLASS()
*
*****************************************************************
*
**     The value specified on the SITE_WARNING parameter
**     specifies whether you want LMSINIT to complete with a
**     return code of 4 whenever SITE records in a license file
**     are skipped (YES), or you want LMSINIT to complete with
**     a return code of 0 (NO). LMS 1.0 always completed with a
**     return code of 4 when LMSINIT detected that there were
**     SITE records in the license file that were not loaded,
**     because their SITE number was not included in the SITE()
**     parameter. If you omit the SITE_WARNING parameter, this
**     behavior will exist in LMS 2.0 as well.  But if you know
**     that you are skipping SITE records, and you want a
**     return code of 0 so that your automated operator program
**     will detect this return code, then include the
**     SITE_WARNING parameter and specify a value of NO.  If
**     this parameter is omitted SITE_WARNING(YES) is used as a
**     default.
**
*
*****************************************************************
*
   SITE_WARNING()          /* YES or NO                 */
*
*****************************************************************
*
**     The next three LMSINIT parameters define master console
**     commands which can automatically be issued by LMSINIT.
**     LMSINIT can complete with a return code of 0, a return
**     code of 4 or a return code of 8 or greater. Each of
**     these three conditions can have a unique console command
**     associated with it. LMSINIT will issue the command that
**     represents the current return code.
**
**     Compuware does not supply any procedures to be started
**     by these commands. It is the customer's responsibility
**     to insure that any PROC specified in a command exists in
**     an appropriate procedure library.
**
**     The examples below issue an OS/390 "START" command. But
**     any valid operator command could be issued as well.
**
**     If no command is specified for a particular return code
**     value, then no command is issued when that return code
**     occurs.
**
**     Any, all or none of the three return code conditions
**     can have a command associated with it.
**
**     Commands can be the same or different for each of the
**     three return codes.
**
**     The commands will be issued as if they were entered at
**     the Master Console, and the Security (RACF, ACF/2 or
**     TOPSECRET) USERID will be the USERID under which LMSINIT
**     is running. Insure that this USERID has the appropriate
**     authority to issue Master Console commands.
**
**     Rules for coding these commands follow:
**
**     If the command contains blanks, then enclose the entire
**     command in single (') or double (") quotes.
**
**     If the command contains blanks and single quotes, then
**     enclose the entire command in double (") quotes.
**
**     If the command contains blanks and double quotes, then
**     enclose the entire command in single (') quotes.
**
**     The single or double quotes will be removed before the
**     command is issued.
**
**     Example:
**
**     SUCCESS_CMD("START someproc,PARM='RC=0'")
**     WARNING_CMD('START othrproc,PARM="RC=4"')
*
*****************************************************************
*
   SUCCESS_CMD()           /* ISSUED IF RETURN CODE = 0 */
   WARNING_CMD()           /* ISSUED IF RETURN CODE = 4 */
   ERROR_CMD()             /* ISSUED IF RETURN CODE > 4 */
```

```
*
*****************************************************************
**
**    The remaining LMSINIT parameters define the E-mail
**    Notification Facility (ENF).
**
**    If you are not using this facility, you may skip the
**    rest of these parameters.
**
*****************************************************************
*
*****************************************************************
**    TCPIP_NAME specifies the name of the TCP/IP protocol
**    stack that is active on this CPU
*****************************************************************
*
   TCPIP_NAME()             /* NAME OF TCP/IP REGION     */
*
*****************************************************************
**    The EMAIL() parameter further limits the number of E-MAIL
**    messages that are automatically generated.   The defaults
**    are shown below. The values that can be coded in place
**    of "WARN" and "NONE" or "FAIL". "NONE" specifies that no
**    E-MAIL messages are to be generated for the product or
**    the option. "FAIL" specifies that only E-MAIL messages
**    reporting failure conditions (i.e. the product or the
**    option is not allowed to execute) are to be generated.
**    "WARN" specifies that both warning and failure messages
**    are to be generated.  E-MAIL messages are still only
**    generated once per product/option per CPU per day, but
**    by specifying the EMAIL parameter, even these messages
**    can be further limited.  This parameter does not affect
**    the messages that products display when warning or
**    failure conditions occur. This parameter ONLY limits the
**    generation of E-MAIL messages.
*****************************************************************
*
   EMAIL(PRODUCT(WARN),OPTION(WARN))  /* limit E-MAIL    */
*
*****************************************************************
**    For Release 3.4 and above of IBM's TCP/IP only:
**    Specify a USERID that the License Management Run Time
**    Environment is to use.
*****************************************************************
*
   TCPIP_USERID()           /* TCP/IP USERID             */
*
*****************************************************************
**    Specify the name, as defined in the domain name server
**    used by this mainframes TCP/IP, of the gateway host
**    used in your network to access the Internet.
**
**    Note: If you specify INTERNET_GATEWAY_NAME, you cannot
**    specify INTERNET_GATEWAY_ADDR, These two operands are
**    alternate methods for specifying the same resource
**    to LMSINIT. One of these two methods MUST be chosen,
**    but both of them cannot be.
*****************************************************************
*
   INTERNET_GATEWAY_NAME()  /* NAME OF INTERNET GATEWAY  */
*
*****************************************************************
**    Specify the address (in dotted decimal notation of the
**    form nnn.nnn.nnn.nnn, where n is a decimal number from
**    1 to 255) of the gateway host used in your network to
**    access the Internet.
**
**    Note: If you specify INTERNET_GATEWAY_ADDR, you cannot
**    specify INTERNET_GATEWAY_NAME. These two operands are
**    alternate methods for specifying the same resource
**    to LMSINIT. One of these two methods MUST be chosen,
**    but both of them cannot be.
*****************************************************************
*
   INTERNET_GATEWAY_ADDR()  /* ADDR OF INTERNET GATEWAY  */
*
*****************************************************************
**    Specify the port number on your Internet gateway host
**    that is used for SMTP traffic destined for the Internet.
**    The default port number that LMSINIT assigns if this
**    operand is omitted is 25, and in all but a very few
**    cases this default is appropriate.
*****************************************************************
*
   INTERNET_GATEWAY_PORT()  /* PORT NUMBER ON GATEWAY    */
```

```
*
*************************************************************
**    Specify the Internet e-mail address (or name) in the
**    form name@institution.type of the individual
**    (or department) that is to be designated as the sender
**    of e-mail messages. This name will appear as the
**    FROM: name on all e-mails automatically generated by
**    the License Management System.
*************************************************************
*
   MAIL_FROM_NAME()         /* INTERNET E-MAIL FROM NAME */
*
*************************************************************
**    Specify the Internet e-mail address (or name) in the
**    form name@institution.type of the individual
**    (or department) that is to receive all automatically
**    generated e-mail messages relating to product
**    licensing errors.
*************************************************************
*
   MAIL_TO_SEC_NAME()       /* INTERNET E-MAIL TO NAME   */
*
*************************************************************
**    Specify the Internet e-mail address (or name) in the
**    form name@institution.type of the individual
**    (or department) that is to receive all automatically
**    generated e-mail messages relating to Compuware
**    License Management software errors (program ABENDs).
*************************************************************
*
   MAIL_TO_ABN_NAME()       /* INTERNET E-MAIL TO NAME   */
*
*
* End of Parameter listing
*
```

Note: For the MAIL_TO_ABN_NAME() parameter, entering an Internet address instructs LMS to send an e-mail to this address for every LMS abend encountered. For Domestic U.S. customers only, enter CSS@COMPUWARE.COM if you want to send these e-mails directly to Compuware License Management System product support.

3. To ensure access to Compuware products is automatically enabled, establish a procedure in your SYS1.PROCLIB to launch LMSINIT as a started task during IPL and IML processing. Consult your site's MVS system programmer for details. Edit the sample proc provided in SLMSCNTL(LMINPROC) (Figure 5-15). Change the dataset names to match those used at your site.

   Note: This proc must run before any other Compuware product procs.

Figure 5-15. Sample LMSINIT Proc

```
//LMSINIT PROC
//LMSSTEP EXEC PGM=LMSINIT,PARM='LANGUAGE=EN'
//STEPLIB   DD DISP=SHR,DSN=USER.LMS.AUTHLOAD        <==== CHANGE DSN
//SYSPRINT  DD SYSOUT=*
//CWLF0000  DD DISP=SHR,DSN=YOUR.VSAM.LICENSE.FILE   <==== CHANGE DSN
//LMSCHKPT  DD DSN=<YOUR.DATA.SET.NAME>,             <==== CHANGE DSN
//             DISP=(MOD,KEEP,KEEP),
//             SPACE=(TRK,0),
//             UNIT=<UNIT>,                          <==== CHANGE UNIT
//             VOL=SER=<VOLSER>                      <==== CHANGE VOLSER
//SYSIN     DD DISP=SHR,
//             DSN=<SMPE.HLQ>.SLMSCNTL(LMINPARM)     <==== CHANGE DSN
```

4. Edit the sample JCL provided in SLMSCNTL(LMINJCL) (Figure 5-16). Change the dataset names to match those used at your site.

Figure 5-16. Sample LMSINIT JCL

```
//*
//* YOUR JOBCARD GOES HERE
//*
//LMSSTEP EXEC PGM=LMSINIT,PARM='LANGUAGE=EN'
//STEPLIB   DD DISP=SHR,DSN=USER.LMS.AUTHLOAD          <==== CHANGE DSN
//SYSPRINT  DD SYSOUT=*
//CWLF0000  DD DISP=SHR,DSN=YOUR.VSAM.LICENSE.FILE     <==== CHANGE DSN
//SYSIN     DD DISP=SHR,
//             DSN=<SMPE.HLQ>.SLMSCNTL(LMINPARM)       <==== CHANGE DSN
```

Step 8. Run LMSINIT

Submit your site's customized LMSINIT JCL created from the sample provided in SLMSCNTL(LMINJCL) (Figure 5-16). This will establish the runtime environment that enables LMS to function.

Step 9. Verify Proper LMS Function

In this step, you will use the LAU to generate a License Verification Report. You can then use that report to verify that the procedures in this chapter were performed correctly and LMS is operating properly.

1. Start the LAU by selecting the ISPF menu item that was added as part of "Step 2. Set up License Administration Utility (LAU)" on page 5-4. The main Compuware License Management screen shown in Figure 5-8 on page 5-9 will be displayed.
2. Type **5** in the Option field and press Enter. The Reports screen (Figure 5-17) will be displayed.

Figure 5-17. Reports Screen

```
---------------------- Compuware License Management 02.00.00 -- Row 1 to 3 of 7
Command ===>                                                  SCROLL ===> PAGE
                              Report Selection
Select "D" for Detail, or "S" for Summary
                           Expire Date   Process DSN
_ License Verification
_ Current Cache Report
_ VSAM License File
_ SAM License File
_ SMF License Records
_ Product Activity
_ Activity Extract

Edit JCL . . . . N
Jobcard:
//Job Card information line 1
//Job Card information line 2
//Job Card information line 3
//Job Card information line 4

Select     Node      Description
  _        Node1     Production
  _        Node2     MVS Test1
  _        Node3     QA Test
```

3. Type S next to License Verification.
4. Type **Y** in the Edit JCL field.
5. Type the information for your jobcard in the four lines of the Jobcard field.
6. Type **S** next to the node(s) on which you want the Report job to run. If no node is selected, the job is submitted without any specific routing.

   Note: If you are running on only one node, do not make a selection here.

7. Press Enter. An Edit JCL screen will be displayed.
8. Make any necessary edits, then enter the END command. A Confirm Submission window will be displayed over the Reports screen.
9. Type **Y** in the Confirm Submission window and press Enter. The JCL to create your License Verification Report will be submitted.
10. Enter the END command. The main Compuware License Management screen (Figure 5-8 on page 5-9) will be displayed.
11. Type **X** in the Option field and press Enter to exit the LAU.
12. Use SDSF to examine the License Verification Report and ensure that the Compuware product being installed is recognized and enabled by LMS.

Step 10. Continue ECC Installation

If the Compuware product you are installing also requires customization of Compuware Shared Services (CSS), and/or Distributed Viewing Support (DVS) which requires that Base Services/HCI be installed, continue with **Chapter 6, "CSS Customization"**. If the Compuware product you are installing does not utilize CSS, return to the manual for that product and complete the installation.

Resolving Problems

If any error messages were returned while performing steps in this chapter, refer to the *Enterprise Common Components Messages and Codes* guide.

If you encountered any other difficulties related to License Management System, please consult the *License Management System User/Reference Guide* for more information. If problems persist, contact Compuware Technical Support.

Chapter 6.
CSS Customization

The procedures in this chapter are used to customize Compuware Shared Services (CSS) and the Compuware Viewing Facility (Compuware/VF).

Note: Only users who are installing Abend-AID XLS with Distributed Viewing Support need to install Base Services/HCI. For information about configuring the Base Services/HCI server, refer to the Abend-AID XLS *User/Reference Guide*.

As shown in Table 6-1, some of the steps in this chapter are required, while others are optional.

Table 6-1. CSS Procedure Requirements

Step Number	Required	Optional
"Step 1. Implement the Security Exit Program (Optional)"		x
"Step 2. Install Customized Translation Tables (Optional)"		x
"Step 3. Make New CSS Load Modules Accessible to Compuware Products (Required)"	x	
"Step 4. Activate Compuware ISPF Dialogs (Required)"	x	
"Step 5. Link Compuware/VF and CSS Utilities Tutorials to Your Main Tutorial Panel (Optional)"		x
"Step 6. Associate Contact Information File with Compuware/VF (Optional)"		x
"Step 7. Establish Access to File-AID (Optional)"		x
"Step 8. Prepare the DDIO File(s)"	x	
"Step 9. Implement the Language Processor JCL (Required)"	x	

Step 1. Implement the Security Exit Program (Optional)

The Security Exit program is an optional user-coded module for establishing security at each site. If you want to ensure full security access to file resources, refer to the "CSS Security Exit" chapter in the Compuware Shared Services User/Reference Guide for detailed information on using the program. A sample Security Exit program is located in SLCXCNTL sample library member CWASSECU.

If the Security Exit program is present, it is called for each member selected. The Security Exit program determines if the user command is permitted, and passes the appropriate return code. If the command is unacceptable for one of the selected members, it will not be executed at all.

Because CSS is used by more than one Compuware product, the Security Exit program is executed for each Compuware product that uses CSS. If you are using multiple Compuware products, you must update your Security Exit program to accommodate those products.

Notes:

1. The sample Security Exit program is for illustration purposes only and **does not** provide any useful functions. You must update it to serve the needs of your site.
2. This Security Exit program is invoked in the MVS environment and does not provide for CICS services.
3. If CICS Abend-AID/FX is installed at your site, or if you are installing CICS Abend-AID/FX, the Security Exit program must be reentrant.

To Implement the Security Exit Program:

Use the following procedure to implement the CWASSECU Security Exit program:

1. Create the Security Exit program. Use the sample Security Exit program as the basis for your exit.

   You must use the program name CWASSECU because other programs reference it and search specifically for this program name.
2. Assemble and linkedit the Security Exit program as a separate load module named CWASSECU. SLCXCNTL sample library member CXJCLSEC contains sample JCL that can be used to assemble and linkedit CWASSECU to the CSS load library. Linkedit CWASSECU into the same load library as CSS.
3. If CICS Abend-AID/FX is installed at your site, the Security Exit program must be reentrant.

Step 2. Install Customized Translation Tables (Optional)

Note: If you omit this step, the default translation tables will be used.

You must perform this step if you are using Japanese language support with Abend-AID.

You may optionally perform this step if you have Abend-AID with the following product facilities:

- Abend-AID
  - Basic Support
  - with Compuware/VF
  - with Compuware/VF and Compuware language processor

When necessary, Abend-AID products dump storage in both hexadecimal and character representation. These storage areas will appear in one of two formats:

- Horizontal dump
- Vertical dump

The character representation for either format is controlled by a default translation table. The default character set is mixed-case English. You can override either, or both, of these default translation tables with customized translation tables. Different customized translation tables can be specified for either dump format. These customized tables may also allow display of certain otherwise non-displayable fields.

If you install vertical and/or horizontal customized translation tables, CWCMTRVT must be used as the load module name for the vertical translation table, and CWCMTRHT must be used as the load module name for the horizontal translation table. A different internal name can be defined to identify a table. The internal table name must be eight characters long. The table length, excluding the table name, must be 256 bytes. Therefore, the total table length is 264 bytes. If an error is made installing a customized translation table, the Compuware default table is used.

The following SLCXCNTL sample library members contain sample translation tables:

CWCMTRVT	Vertical translation table for uppercase English.
CWCMTRHT	Horizontal translation table for uppercase English.
CWCMTRVE	Vertical translation table for mixed-case English. This is the default vertical translation table that will be used if you choose not to install a customized vertical translation table.
CWCMTRHE	Horizontal translation table for mixed-case English. This is the default horizontal translation table that will be used if you choose not to install a customized horizontal translation table.
CWCMTRVU	Vertical translation table for mixed-case English with the Euro Character X‘9F’ included.
CWCMTRHU	Horizontal translation table for mixed-case English with the Euro character X’9F’ included.

CSS SLCXCNTL sample library member CXJCLTRT contains the sample JCL to assemble and link-edit customized translation tables.

Step 3. Make New CSS Load Modules Accessible to Compuware Products (Required)

You must perform this step.

Ensure that the new CSS load modules can be accessed by Compuware products by using one or more of the following methods. Regardless of the method chosen, the CSS load library (SLCXLOAD or a copy of it) should be concatenated **in front** of any Compuware product library(ies). The method you choose will depend on the Compuware product(s) you have at your site.

- Compuware recommends that you place the CSS load library in the link list. This will provide access to the CSS load modules for TSO users and batch jobs that need CSS (for example, language processors, CSS utilities, and/or Abend-AID). This also minimizes any JCL changes that might be needed for a CSS upgrade. (A refresh of the link list is required to activate this change.) Refer to "APF Authorization" on page 2-7 for additional information and cautions about APF authorizations.
- If you have XPEDITER/CICS installed at your site, place the CSS load library in the DFHRPL concatenation for each CICS region ahead of the XPEDITER/CICS libraries.
- You may also place the CSS load library in the STEPLIB or ISPLLIB for individual TSO logon PROCs, and in the STEPLIB or JOBLIB for batch jobs. These library specifications will override the CSS load library specified in the link list. This is useful for testing a new version of CSS prior to placing it into the link list for production use.

Notes:

1. If you have previously installed CSS and referenced the load library in batch jobs, compile PROCs, CLISTs, or logon PROCs, CSS will be loaded from the STEPLIB, ISPLLIB, or JOBLIB where it is specified rather than from the link list. Compuware recommends that you review these locations and remove or modify the CSS load library DDs as appropriate. If you have any Compuware product that uses CSS from a STEPLIB, ISPLLIB, or JOBLIB concatenation, you should include the CSS load library in front of the Compuware product in that library concatenation.

   Beginning with CSS 6.0, you have the ability to dynamically allocate the CSS load library as part of the ISPLLIB concatenation. Please refer to Option 1 below for more information.

2. For CICS Abend-AID/FX, specify the CSS load library name in the server JCL. Refer to the CICS Abend-AID/FX *Installation and Customization Guide* for more information.
3. If you use the Abend-AID XLS Distributed Viewing Support feature, you must make the Base Services/HCI Load Library (SKMPLOAD) available to the Compuware Viewing Facility either in your TSO logon PROC or CLIST, or dynamically when you start the Viewing Facility. See "Option 1 -- Dynamically Allocate Libraries when Compuware/VF or CSS Utilities is Executed" on page 6-5 for more information.

Step 4. Activate Compuware ISPF Dialogs (Required)

You should perform this step if you plan to use the Compuware Viewing Facility and/or CSS Utilities.

Compuware ISPF dialogs include the Compuware Viewing Facility (Compuware/VF) and CSS Utilities. This section covers two different ways to activate the Compuware/VF and CSS Utilities libraries—dynamically, or at TSO startup.

Dynamically allocating the libraries is discussed in Option 1 below. Allocating the libraries at TSO startup is discussed in "Option 2 -- Allocate the Libraries at TSO Startup" on page 6-6. For more information about Compuware/VF or CSS Utilities functionality, see the *Compuware Shared Services User/Reference Guide*.

Option 1 -- Dynamically Allocate Libraries when Compuware/VF or CSS Utilities is Executed

To dynamically allocate the Compuware/VF and CSS Utilities libraries, you must determine how to make the REXX EXECs and CLISTs accessible.

CSS provides sample REXX EXECs and CLISTs for invoking Compuware/VF and CSS Utilities in SLCXCNTL sample library. You can allocate this library at TSO logon time, or make the CLIST or REXX EXEC accessible in a user-defined library allocated at TSO logon.

IMPORTANT

If CSS REXX EXECs or CLISTs are currently used to invoke Compuware/VF, you must replace them with the latest REXX EXECs or CLISTs in order to use the new Abend-AID features.

Complete the following steps:

1. Determine whether to use REXX EXECs or CLISTs.

   Compuware recommends using REXX EXECs; you may find them easier to customize and maintain.

2. Determine whether Japanese and English language support is required, or just English language support.

3. If using REXX EXECs, choose the appropriate REXX EXEC for Compuware/VF and the appropriate REXX EXEC for CSS Utilities using the following table as a guide. If using CLISTs, select one CLIST for Compuware/VF and one CLIST for CSS Utilities.

4. Change the library names in the REXX EXECs or CLISTs selected in Step 3 to match your site standards.

   Note: Effective with CSS Release 7.9, an optional ISPSLIB can be used with On-the-Fly processing. See Appendix C in the *Compuware Shared Services User/Reference Guide* for details about On-the-Fly processing. You may use an existing skeleton library allocated at TSO logon time, or allocate one at viewer startup by updating the REXX EXECs or CLISTs selected in "Step 3. Make New CSS Load Modules Accessible to Compuware Products (Required)" on page 6-4

Use this REXX EXEC	Or this CLIST	For
CWVFREXE	CWVFCLSE	Compuware/VF, English-only language support, view Abend-AID reports or run utilities
CWVFREXJ	CWVFCLSJ	Compuware/VF, English and Japanese language support, view Abend-AID reports or run utilities

Use this REXX EXEC	Or this CLIST	For
CWUTREXE	CWUTCLSE	CSS Utilities, English-only language support, set up or run XPEDITER utilities
CWUTREXJ	CWUTCLSJ	CSS Utilities, English and Japanese language support, set up or run XPEDITER utilities

Note: When testing a new install of Abend-AID XLS and the new Abend-AID XLS load library has not yet been placed in the link list, it must be listed in CWVFREXE or CWVFCLSE (and/or the Japanese versions) under the ISPLLIB concatenation. Doing this will prevent text merge messages when viewing an Abend-AID XLS report.

5. Allocate the CSS SLCXCNTL library at TSO logon time, or copy the edited REXX EXECs/CLISTs to a library allocated at TSO logon:

   – **To allocate the CSS SLCXCNTL library at TSO logon**

   Add the library to the SYSPROC concatenation in the TSO logon PROC.

   – **To make specific REXX EXECs/CLISTs accessible in a library allocated at TSO logon**

   Copy the REXX EXECs or CLISTs to an existing SYSPROC library allocated at TSO logon.

6. Users can invoke either Compuware/VF or CSS Utilities by selecting an option from an ISPF panel (recommended), or by executing a REXX EXEC or CLIST.

   – **To enable users to select an ISPF menu option**

   Add one or more of the following lines to an ISPF panel. Note that 'o' represents the option to be entered at the menu command prompt.

To invoke Compuware/VF	o,'CMD (%*Compuware/VF CLIST name* **or** *Compuware/VF REXX EXEC name*)
To invoke CSS Utilities	o,'CMD (%*CSS Utilities CLIST name* **or** *CSS Utilities REXX EXEC name*)

   – **To invoke Compuware/VF or CSS Utilities by executing a REXX EXEC or CLIST**

   Enter the appropriate command at the TSO command prompt:

To invoke Compuware/VF	TSO %*Compuware/VF CLIST name* **or** *Compuware/VF REXX EXEC name*
To invoke CSS Utilities	TSO %*CSS Utilities CLIST name* **or** *CSS Utilities REXX EXEC name*

Option 2 -- Allocate the Libraries at TSO Startup

Allocating the load, panel, and message libraries at TSO logon simplifies maintenance. All TSO users have access to Compuware/VF and CSS Utilities after logging on to TSO, and upgrades that affect all TSO users can be made in one central location. Allocating these libraries at TSO logon may be inappropriate, however. In that case, dynamically allocating the libraries may be a better solution.

To provide access to the load, panel, and message modules at TSO logon time, complete the following steps.

1. Concatenate each library as shown in the following table:

Table 6-2. Allocating CSS Libraries

Load library SLCXLOAD	Place the CSS load library in the link list (see "Step 3. Make New CSS Load Modules Accessible to Compuware Products (Required)" on page 6-4 for instructions) OR Place a DD statement for the load library in the concatenation for STEPLIB or ISPLLIB in your TSO logon PROC OR Execute a CLIST or REXX EXEC at logon time that will concatenate the load library to the appropriate DD statement before starting ISPF
Load library SKMPLOAD Base Services/HCI — required only for Abend-AID XLS Distributed Viewing Support	Place the Base Services/HCI load library in the link list (see "Step 3. Make New CSS Load Modules Accessible to Compuware Products (Required)" on page 6-4 for instructions) OR Place a DD statement for the load library in the concatenation for STEPLIB or ISPLLIB in your TSO logon PROC OR Execute a CLIST or REXX EXEC at logon time that will concatenate the load library to the appropriate DD statement before starting ISPF
Message library SLCXMENU	Concatenate the library to ISPMLIB in the TSO logon PROC OR Concatenate the library to ISPMLIB in a CLIST or REXX EXEC before starting ISPF
Panel library SLCXPENU	Concatenate the library to ISPPLIB in the TSO logon PROC OR Concatenate the library to ISPPLIB in a CLIST or REXX EXEC before starting ISPF
EXEC library SLCXEXEC	Concatenate the library to SYSPROC in the TSO logon PROC OR Concatenate the library to SYSPROC in a CLIST or REXX EXEC before starting ISPF
Skeleton Library ISPSLIB	Optional library for On-the-Fly processing. See Appendix C in the *Compuware Shared Services User/Reference Guide* for details. Concatenate the library to ISPSLIB in the TSO logon PROC OR Concatenate the library to ISPSLIB in a CLIST or REXX EXEC before starting ISPF.

Notes:

a. Replace any existing CSS libraries in these concatenations.

b. The low level qualifiers are listed below their respective library names.

c. When testing a new install of Abend-AID and the new Abend-AID load library has not yet been placed in the link list, it must be added as explained in the instructions for the SLCXLOAD library. Doing this will prevent text merge messages when viewing an Abend-AID report.

2. Add one or more of the following lines to an ISPF panel. Note that ‘o’ represents the option to be entered at the ISPF menu command prompt.

To invoke Compuware/VF	o,'PGM(CWDDSUTL) NEWAPPL(AAUT) PARM(SPF)'
To invoke CSS Utilities	o,'PGM(CWDDIUFE) NEWAPPL(AAUT)'

Step 5. Link Compuware/VF and CSS Utilities Tutorials to Your Main Tutorial Panel (Optional)

CSS SLCXCNTL sample library contains two sample tutorial main menu panels for ISPF and ISPF/PDF:

- CXTUTOR
- CXR00003

To link the Compuware/VF and CSS Utilities tutorial panels to your tutorial main menu panel (TTUTOR or ISR00003), use the applicable member as an example.

Step 6. Associate Contact Information File with Compuware/VF (Optional)

The Contact Information feature enables Compuware/VF users to associate contact information (for example, the phone number and e-mail address of an off-duty programmer) with a particular job name. In order to use this feature, the contact information file must be associated with Compuware/VF.

Complete the following steps:

1. Use the following members from the CSS SLCXCNTL sample library:
   - CXCIVSAM
   - CXCIREG
2. Edit member CXCIVSAM and modify the COMPWARE.CONTACT.INFO dataset name to conform to your installation's naming standards.
3. Execute CXCIVSAM to allocate the VSAM dataset that will store contact information.
4. Edit member CXCIREG and change the CDSN symbolic parameter to the dataset name used in Step 2.
5. Set the CXLOAD symbolic parameter to the CSS load library dataset name.
6. Set the CXCNTL symbolic parameter to the CSS SLCXCNTL sample JCL dataset name.
7. (Optional) Specify the UNIT and ASSEMBLER symbolic parameters.
8. Execute CXCIREG to register the contact information dataset.
9. To populate the Contact Information file, refer to the *Compuware Shared Services User/Reference Guide* "Working with Contact Information" section in the Compuware Viewing Facility chapter.

If you wish to use a Contact Information dataset created prior to CSS 7.8 (PTF level CXF0168), it must be converted into the new extended format. The CSS SLCXCNTL sample library contains the CXCIVSAM and CXCICONV members.

1. Edit the CXCIVSAM member and change the dataset name to a new name. Execute the job.
2. Edit the CXCICONV member. Change the CIDSN symbolic parameter to point to the original contact information dataset.
3. Change the CODSN symbolic parameter to point to the new dataset you just allocated.
4. Execute CXCICONV. Follow the instructions for step 9 above and adjust the Contact Information data if necessary.

Step 7. Establish Access to File-AID (Optional)

File-AID users may now access File-AID/MVS, File-AID for DB2, or File-AID for IMS directly from Compuware/VF using simple line commands. Installation requirements for each version of File-AID are listed below.

File-AID/MVS Users

To access File-AID/MVS from Compuware/VF, complete the following steps:

1. Ensure that File-AID/MVS 8.5 or more current is installed.
2. Either:
   - Allocate the File-AID/MVS libraries at TSO logon

     OR
   - Locate the CWVFCLSE CLIST or CWVFREXE REXX EXEC in the SLCXCNTL sample library. Uncomment the lines labeled **FA/MVS PANEL LIB**, **FA/MVS MESSAGE**, **FA/MVS LOADLIB**, and **FA/MVS LOADLIB** and replace the existing library names with your File-AID/MVS library names.

File-AID for DB2 Users

To access File-AID for DB2 from Compuware/VF, complete the following steps:

1. Ensure that File-AID for DB2 V3R9 or more current is installed.
2. Locate the CWVFDB2 and FDXTRN01 CLISTs in the SLCXCNTL sample library.
3. Move these CLISTs to a library allocated at TSO logon.

   If Compuware/VF is started by executing CWVFCLSE or CWVFREXE you can place these CLISTs into the SLCXEXEC library.
4. Locate the following line in CLIST CWVFDB2:

   ```
   /* SET &FADB2CL = '????????.FADB2.ISRCLIB'
   ```

   Replace this dataset name with the name of the File-AID for DB2 CLIST library.
5. (Optional) To dynamically allocate the File-AID for DB2 ISPF and CLIST libraries when a user accesses File-AID from Compuware/VF, set the LIBDEF PROC variable in CLIST CWVFDB2 to **Y**.

File-AID for IMS Users

To access File-AID for IMS from Compuware/VF, the following conditions must be met:

- File-AID for IMS 4.7 with product update F1 must be installed.
- File-AID for IMS CLISTs FIALLOC and FIFREE must exist in a CLIST allocated to the user's SYSPROC DD.

Step 8. Prepare the DDIO File(s)

You must perform this step if you are a new CSS user or have never created DDIO files and now need to use them. If you are an Abend-AID XLS user and are installing or migrating to Release 9.4 or later you will need to allocate a Report Shared Directory and one or more Report databases.

CSS DDIO files are used to store diagnostic reports, transaction reports, and source listings. The term *DDIO file* is a generic term used to refer to the datasets that store the reports and listings from Compuware Products that use CSS. This step describes the various types of DDIO files available and who needs to allocate which type.

Types of DDIO Files

There are two main categories of DDIO files: report or source listing files and report or source listing *databases* which are grouped into pools that are attached to Shared Directories. CSS uses the Shared Directory architecture to manage the available pool space.

Within these two categories of DDIO files, there are three different types of DDIO files characterized by the type of information that is stored in them: abend reports, source listings, and CICS transaction reports.

Note: If you use the Shared Directory architecture for your DDIO files, different database types cannot be attached to the same Shared Directory (for example, you cannot attach a source listing database to a report shared directory, you cannot attach a transaction database and report database to the same Shared Directory, etc.).

Note: Non-database report or source listing files cannot be attached to a Shared Directory until they have been converted. See the "Batch File Utility CWAASDUT" and the "Batch File Utility CWDDLPUT" chapters in the *CSS User/Reference Guide* for more information.

1. Abend Reports:
   a. **Report Shared Directories and Databases.** This type of file is used to store Abend Reports created by Abend-AID XLS. *This Report Shared Directory architecture is only supported for Abend-AID XLS Release 9.4 and later.* Report databases are attached to a report shared directory. The report databases are managed as a POOL of space belonging to the report shared directory to which they are attached.
   b. **Report DDIO files.** This type of DDIO file is used to store Abend Reports created by Abend-AID XLS. Prior to Abend-AID Release 9.4, this was the only type of report file available. For compatibility purposes, Abend-AID XLS Release 9.4 can also use this type of report file. A conversion utility is available to run when the user is ready to migrate the report file to the report database format.
2. Source Listings:
   a. **Source Listing Shared Directories and Databases.** This type of DDIO file is used to store compiler source listings for use by various Compuware Products. Source listing databases are attached to a source listing shared directory. The source listing databases are managed as a POOL of space belonging to the source listing shared directory to which they are attached. *The Source Listing Shared Directory architecture can now be used with all supported language compilers. However, XPEDITER product users and* CICS Abend-AID/FX *users should see notes and exceptions below.*
   b. **Source Listing DDIO Files.** This type of DDIO file is used to store compiler source listings. A conversion utility is available to run when the user is ready to migrate the listing file to the source listing database format.

The same source listing file, database, or source listing member can be used by multiple Compuware Products. It is not necessary to create duplicate source listing files, databases, or source listing members for each Compuware Product.

3. **Transaction Databases and Shared Directory for** CICS Abend-AID/FX**.** Transaction databases are used by CICS Abend-AID/FX to store CICS transaction dumps. CICS Abend-AID/FX catalogs both CICS transaction dumps and imported Region Dumps into the shared directory.

The following table lists the types of DDIO files created by the CSS utilities and which products use/require them:

Table 6-3. DDIO File Formats

Compuware Product	Report File	Source Listing File	Transaction Databases & Shared Dir.	Report Shared Dir. & Database	Source Listing Shared Dir.	Source Listing Database
Abend-AID XLS	x	x		x	x	x
CICS Abend-AID/FX		x	x			x
XPEDITER/TSO		x			x[1] See Note	x
XPEDITER/IMS		x			See Note	x
XPEDITER/CICS		x			See Note	x

[1] If using XPEDITER/TSO with the C Language Processor, VisualAge for PL/I, or Enterprise PL/I and you use Long Program Names, the Source Shared Directory must be used.

Note: The XPEDITER CICS and CICS Abend-AID/FX products **do not** support Source Shared Directories. However, individual source listing databases **can** be used by all CSS-enabled Compuware products if the databases are explicitly referenced. See footnote 1 above for exceptions.

Refer to the "Source Listing Databases and Shared Directories" section of Chapter 2 in the *Compuware Shared Services User/Reference Guide* for more information.

File Access Methods

You may use either sequential or VSAM files for DDIO files. Shared directories must be VSAM files. If you are using VSAM files for your DDIO files, you can copy them using the IDCAMS REPRO command. You cannot, however, change any of the file attributes such as CISIZE. Allocation and format requirements, along with the steps for preparing the various DDIO files, will vary according to the type of file. Source listing files can be shared between all Compuware products that use CSS, while report files, transaction databases, work files, and profile files, cannot.

CAUTION:

CWDDSUTL, CWFXSDUT, CWAASDUT, and CWDDLPUT are the only utilities recommended for use with Compuware DDIO files.

File-AID/MVS and other VSAM-related vendor products (including vendor packages that compress or enhance buffer management on VSAM files) should NOT be used against Compuware DDIO files. Doing so can result in corrupting the contents of the DDIO file. See "Chapter 7. CSS: Allocating and Formatting DDIO Files" or the *Compuware Shared Services Installation and Customization Guide* for more information.

Step 8a. Allocate and Format Abend-AID Report Shared Directory(s) and Database(s) for Abend-AID XLS Release 9.4 and Later Users

This step should only be performed by Abend-AID XLS users who have installed, are installing, or are migrating to Abend-AID XLS Release 9.4 or later. The purpose of this step is to allocate and format a Report Shared Directory and one or more Report Databases attached to that Shared Directory. Note that Abend-AID XLS 9.4 is the minimum release able to use these types of DDIO file.

This step can be performed now, or during the Abend-AID XLS installation.

Create the Report Shared Directory

Use the sample JCL in the CSS SLCXCNTL dataset to allocate and format the Report Shared Directory (member is CXALLDAA). For additional options refer to the "Batch File Utility CWAASDUT" chapter in the *Compuware Shared Services User/Reference Guide*. The shared directory CREATE command will both ALLOCATE and FORMAT the shared directory. If the file currently exists, you must specify parameter REALLOCATE=YES or the allocate will fail with a duplicate file error.

Create the Report Database(s)

You must have already created the Report Shared Directory before attempting to create the Report Database. You **can** create both in the same job as long as the Report Shared Directory create is done first. The database CREATE command will both ALLOCATE and FORMAT the database. If the file currently exists, you must specify parameter REALLOCATE=YES or the allocate will fail with a duplicate file error.

VSAM Report Database(s) can be created using the sample JCL in the CSS SLCXCNTL dataset (member is CXALLVAA).

Sequential Report Database(s) can be created using the sample JCL in the CSS SLCXCNTL dataset (member is CXALLBAA).

Step 8b. Allocate and Format Abend-AID XLS Report Files for pre-Abend-AID XLS Release 9.4 Users

You **must** perform this step if you are an Abend-AID XLS user and did **not** perform "Step 8a. Allocate and Format Abend-AID Report Shared Directory(s) and Database(s) for Abend-AID XLS Release 9.4 and Later Users", **and** *do not* have an existing Report File.

Allocate the Report File(s)

VSAM Report files can be allocated using the sample JCL in the CSS SLCXCNTL dataset (member is CXALLVS, parm member is CXALDDAA).

Sequential Report files can be allocated using the sample JCL in the CSS SLCXCNTL dataset member CXALLDS.

Format the Report File(s)

After the report file(s) have been allocated, they must be formatted before they can be used. The formatting process initializes the directory structure and specifies the file options.

CSS SLCXCNTL library member CXFMTDS contains sample JCL to format a report file with appropriate defaults for the sample files allocated using members CXALLVS or CXALLDS. Change CXPARM to specify CXAARPT as the member containing the FORMAT control statement for the report file being formatted. You can optionally specify your own customized member in place of CXAARPT.

Step 8c. Allocate and Format Source Listing Shared Directory(s) and Database(s)

This step may be performed by all users who use Source Listing support. Please note the exceptions for XPEDITER and CICS Abend-AID/FX users above.

Create the Source Listing Shared Directory.

Use the sample JCL in the CSS SLCXCNTL dataset to allocate and format the Source Listing Shared Directory (member is CXALLDLP). For additional options refer to the "Batch File Utility CWDDLPUT" chapter in the *Compuware Shared Services User/Reference Guide*. The shared directory CREATE command will both ALLOCATE and FORMAT the shared directory. If the file currently exists, you must specify parameter REALLOCATE=YES or the allocate will fail with a duplicate file error.

Create the Source Listing Database(s)

You must have already created the Source Listing Shared Directory before attempting to create the Source Listing Database. You **can** create both in the same job as long as the Source Listing Shared Directory create is done first. The database CREATE command will both ALLOCATE and FORMAT the database. If the file currently exists, you must specify parameter REALLOCATE=YES or the allocate will fail with a duplicate file error.

VSAM Source Listing Database(s) can be created using the sample JCL in the CSS SLCXCNTL dataset (member is CXALLVLP).

Sequential Source Listing Database(s) can be created using the sample JCL in the CSS SLCXCNTL dataset (member is CXALLBLP).

Step 8d. Allocate and Format Source Listing Files

You must perform this step if you did **not** perform "Step 8c. Allocate and Format Source Listing Shared Directory(s) and Database(s)" **and** you do not have an existing source listing file or database.

Allocate the Source Listing File(s)

VSAM Source Listing files can be allocated using the sample JCL in the CSS SLCXCNTL dataset (member is CXALLVS, parm member is CXALDDSL).

Sequential Source Listing files can be allocated using the sample JCL in the CSS SLCXCNTL dataset member CXALLDS.

Format the Source Listing File(s)

After the source listing file(s) have been allocated, they must be formatted before they can be used. The formatting process initializes the directory structure and specifies the file options.

CSS SLCXCNTL library member CXFMTDS contains sample JCL to format a source listing file with appropriate defaults for the sample files allocated using members CXALLVS or CXALLDS. Change CXPARM to specify CXFMTLST as the member containing the FORMAT control statement for the source listing file being formatted. You can optionally specify your own customized member in place of CXFMTLST.

Step 8e. Allocate and Format CICS Abend-AID/FX Transaction Shared Directory(s) and Database(s)

Normally this would be performed during the CICS Abend-AID/FX install procedure. For additional information, see the CICS Abend-AID/FX *Installation and Customization Guide*.

Create the CICS Abend-AID/FX Shared Directory (Allocate & Format)

Use the sample JCL to create the shared directory. CSS SLCXCNTL sample library member CXALLMC contains the sample JCL for creating a CICS Abend-AID/FX shared directory.

Create the CICS Abend-AID/FX Database (Allocate & Format)

You must create the shared directory before proceeding to this step. Use one of the following two sets of sample JCL to create each CICS Abend-AID/FX database as needed:

VSAM CICS Abend-AID/FX **Database:** CSS SLCXCNTL sample library member CXALLVSD contains sample JCL for creating a CICS Abend-AID/FX VSAM transaction database.

Sequential CICS Abend-AID/FX **Database:** CSS SLCXCNTL sample library member CXALLBSD contains sample JCL for creating a CICS Abend-AID/FX sequential transaction database.

Step 9. Implement the Language Processor JCL (Required)

See the sub-steps below to determine when you must perform this step.

The language processors support two methods of processing your programs: preprocessing and postprocessing. Both of these choices may involve modifications to your compile JCL.

You may use the preprocessor or postprocessor (whichever is best for your site as needed.)

This section discusses the benefits of the preprocessor and the postprocessor and describes when to use them. See the Compuware Shared Services User/Reference Guide for more information on determining when to use the pre- or postprocessor.

Refer to "CSS Language Processor (LP)" on page 2-10 for descriptions of preprocessing and postprocessing and a summary of the benefits provided by each method.

To use the language processor, you must have previously completed installation of the CSS load library and also prepared a DDIO file. For each language you are using, you should modify the corresponding compiler procedure to provide for the type of processing you will be implementing.

When running the language processor as a preprocessor, you can specify assembler/compiler options in the CWPPRMO dataset or in the PARM statement passed to the preprocessor. The CWPPRMO is an input dataset that contains the language processor compiler options. Depending on the method used, it may be necessary to include the LANGPARM command with these options.

If you are currently using the postprocessor, and you will be converting to the preprocessor, you are not required to spool your SYSPRINT to a temporary dataset.

Note: For each of the following steps, it is possible that you will receive a return code other than zero. Should this occur, note the accompanying language processor error message and refer to the *Enterprise Common Components Messages and Codes* guide for further information.

Step 9a. Modify the JCL to Run the COBOL Language Processor

You must perform this step if either of the following is true:

- You are installing XPEDITER/CICS for the first time.
- You are installing Abend-AID for the first time or you are upgrading a current Abend-AID installation with Abend-AID XLS **and** you are licensed for the Compuware COBOL language processor.

You may optionally perform this step if you are installing CICS Abend-AID/FX or XPEDITER/TSO **and** you are licensed for the Compuware COBOL language processor. XPEDITER/TSO customers can either perform this step now in batch mode, or online as part of the XPEDITER/TSO installation procedure.

This step contains instructions for both the COBOL language preprocessor and postprocessor. You should follow the instructions that apply to the type of processing you will be performing.

Preprocessor

SLCXCNTL sample library member CXCOBPRE contains sample JCL necessary to run the COBOL language preprocessor.

Modify the compile step of your COBOL JCL:

- On the EXEC statement, change the name of your compiler to CWPCMAIN and add the following to any existing COBOL compiler options:

```
LANGUAGE(compiler version)
```

Notes:

a. The LANGUAGE command may be specified in the CWPPRMO command stream rather than on the EXEC statement.

b. The preprocessor accepts VSCOBOLII, VSCOBOLIIREL3, VSCOBOLIIREL4, COBOL/370, COBOL/390, and COBOL/MVS as being equivalent because the program name is the same. The compiler that is actually executed depends on the library specified in the STEPLIB, JOBLIB, and/or LINKLIST concatenation.

- Add the CSS SLCXLOAD DSN to the STEPLIB or JOBLIB concatenation if it is not in the link list.
- Add the CWPDDIO DD statement to specify the name of the source listing file (DDIO file) you will be using.
- Determine the language processor options you will use as input and add them to the EXEC parameter, or add a CWPPRMO DD statement that points to the appropriate options. Sample options are contained in SLCXCNTL sample library members CXLPCOBB (for batch programs) and CXLPCOBC (for CICS programs) to provide the smallest output to the source listing file (for minimizing space) and generate a printout of the COBOL enhanced listing. When using these options, a printout of the listing from the DDIO file contains only the source statements. The XREF and other portions of the listing are not written to the DDIO file. If you use the sample options, note the following:
  - If you do not have XPEDITER/TSO, remove DDIO(OUTPUT(FIND)).
  - If the program will be used by XPEDITER/CICS or CICS Abend-AID/FX, use the options in sample CXLPCOBC.
- The SYSPRINT file will be LRECL=133, RECFM=FBA.

Postprocessor

You can use the postprocessor to process compile listings that were previously stored, or you can modify your COBOL compile JCL to pass the compiler listing to the postprocessor as described below.

- Check the COBOL compiler options.
- Change the COBOL compiler JCL to write the compiler listing (SYSPRINT) to a sequential file that can be used as input to the COBOL language postprocessor. A sample of the necessary JCL is shown below.

```
//SYSPRINT DD UNIT=SYSDA,SPACE=(TRK,(25,20)),DCB=BLKSIZE=16093,
//            DISP=(MOD,PASS)         <===== CA-OPTIMIZER NEEDS 'MOD'
```

  The above example is a temporary file. You can save the JCL as a permanent file to be reused as shown in the example below:

  //SYSPRINT DD DSN=CX,STORED.LISTINGS(MEMBER)

- The SYSPRINT file will be LRECL=133, RECFM=FBA.
- Add SLCXCNTL sample library member CXCOB1 as the postprocessor step following your COBOL compile step.
- Determine the language processor options you will use as input and add them to the EXEC parameter, or add a CWPPRMO DD statement that points to the appropriate options. Sample options are contained in SLCXCNTL sample library members CXLPCOBB (for batch programs) and CXLPCOBC (for CICS programs) to provide the

smallest output to the source listing file (for minimizing space) and generate a printout of the COBOL listing. When using these options, a printout of the listing from the DDIO file will contain only the source statements. The XREF and other portions of the listing will not be written to the DDIO file. If you use the sample options, note the following:

- If you do not have XPEDITER/TSO, remove DDIO(OUTPUT(FIND)).
- If the program will be used by XPEDITER/CICS or CICS Abend-AID/FX, use the options in sample CXLPCOBC.

The SLCXCNTL sample library members that are supplied for the COBOL language postprocessor invoke the language processor in different ways:

Table 6-4. SLCXCNTL Sample Library Members for COBOL Language Processor

Member	Purpose
CXCOB1	Processes compiler listings through the language postprocessor. Add this JCL after your COBOL compile step and before your link edit step (if one exists).
CXCOB2	Postprocesses compiler listings stored from a previous compile job. Use this JCL to process listings that have been previously compiled and stored in machine-readable format.
CXCOB99	Compiles COBOL programs and feeds the compiler output into the COBOL postprocessor.

- The CWPPRTO file will be LRECL=133, RECFM=FBA.
- The CWPLOAD file must be allocated using DISP=OLD or DISP=SHR.
- The object module that is processed by the postprocessor must be used as input to the linkedit step.

Step 9b. Modify the JCL to Run the PL/I Language Processor

You must perform this step if the following is true:

- You are installing Abend-AID with Compuware/VF, **or** XPEDITER/CICS, **or** XPEDITER/TSO **and** you are licensed for the Compuware PL/I language processor.

This step contains instructions for both the PL/I language preprocessor and postprocessor. You should follow the instructions that apply to the type of processing you will be performing.

Preprocessor

SLCXCNTL sample library member CXPLIPRE contains sample JCL necessary to run the PL/I language preprocessor.

Modify the compile step of your PL/I JCL:

- On the EXEC statement, change the name of your compiler to CWPPMAIN and add the following to any existing PL/I compiler options:

```
LANGUAGE(compiler version)
```

  Refer to the *Compuware Shared Services User/Reference Guide* for a list of the compiler versions that can be specified with the LANGUAGE command. The default compiler for PL/I is the compiler currently in use on your system.

  Note: The LANGUAGE command may be specified in the CWPPRMO command stream rather than on the EXEC statement.

- Add the CSS SLCXLOAD DSN to the STEPLIB or JOBLIB concatenation if it is not in the link list.

- Add the CWPDDIO DD statement to specify the name of the source listing file (DDIO file) you will be using.
- Determine the language processor options you will use as input and add them to the EXEC parameter, or add a CWPPRMO DD statement that points to the appropriate options. Sample options are contained in SLCXCNTL sample library member CXLPPLI to provide the smallest output to the source listing file (for minimizing space). If you use these options, and you will be using the PL/I language preprocessor, add LANGUAGE(xxx) to the EXEC PARM or to the CWPPRMO DD statement.

See the *Compuware Shared Services User/Reference Guide* for more information on this JCL and PL/I language processor options.

Postprocessor

You can use the postprocessor to process compile listings that were previously stored, or you can modify your PL/I compile JCL to pass the compiler listing to the postprocessor as described below.

- Check the PL/I compiler options.
- Change the PL/I compile JCL to write the compiler listing (SYSPRINT) to a sequential file for use as input to the PL/I language postprocessor. A sample of the necessary JCL is shown below.

```
//SYSPRINT DD UNIT=SYSDA,SPACE=(CYL,(100,10)),DCB=BLKSIZE=19069,
//            DISP=(MOD,PASS)
```

  The above example is a temporary file. You can save the JCL as a permanent file to be reused as shown in the example below:

  //SYSPRINT DD DSN=CX,STORED.LISTINGS(MEMBER)

- The SYSPRINT file will be LRECL=133, RECFM=FBA.

Table 6-5. SLCXCNTL Sample Library Members for PL/I Language Processor

Member	Purpose
CXPLI	Processes compiler listings through the language postprocessor. Add this JCL after your PL/I compile step and before your link edit step (if one exists). CXPLI can be customized to meet your site's requirements.
CXPLI2	Postprocesses compiler listings stored from a previous compile job. Use this JCL to process listings that have been previously compiled and stored in machine-readable format.

- The CWPLOAD file must be allocated using DISP=OLD or DISP=SHR.
- The object module that is processed by the postprocessor must be used as input to the linkedit step.

See the *Compuware Shared Services User/Reference Guide* for more information on this JCL and PL/I language processor options.

Step 9c. Modify the JCL to Run the Assembler Language Processor

You must perform this step if you are licensed for the Assembler language processor option for any product.

You can optionally perform this step if you are installing XPEDITER/TSO **and** you are licensed for the Compuware Assembler language processor. XPEDITER/TSO customers can either perform this step now in batch mode, or online as part of the XPEDITER/TSO installation procedure.

This step contains instructions for both the Assembler language preprocessor and postprocessor. You should follow the instructions that apply to the type of processing you will be performing.

SLCXCNTL sample library member CXASMPRE contains sample JCL necessary to run the Assembler language preprocessor.

Preprocessor

Modify the assembly step of your Assembler JCL:

- On the EXEC statement, change the name of your compiler to CWPAMAIN and add the following to any existing assembler options:

  ```
  LANGUAGE(assembler version)
  ```

  Refer to the *Compuware Shared Services User/Reference Guide* for a list of the assembler versions that can be specified with the LANGUAGE command. The default assembler is Assembler H.

  Note: The LANGUAGE command may be specified in the CWPPRMO command stream rather than on the EXEC statement.

- Add the CSS SLCXLOAD DSN to the STEPLIB or JOBLIB concatenation if it is not in the link list.
- Add the CWPDDIO DD statement to specify the name of the source listing file (DDIO file) you will be using.
- Determine the language processor options you will use as input and add them to the EXEC parameter, or add a CWPPRMO DD statement that points to the appropriate options. Sample options are contained in SLCXCNTL sample library member CXLPASM to provide the smallest output to the source listing file (for minimizing space). When using these parameters, a printout of the listing from the DDIO file will contain only the source statements. The XREF will not be written to the DDIO file. If you use these options and you will use the Assembler language preprocessor, add LANGUAGE(xxx) to the EXEC PARM or to the CWPPRMO DD statement.
- Specify the compiler (and release) you will use by including it as a parameter of the LANGUAGE command in your EXECUTE parameter or CWPPRMO DD statement.

Refer to the *Compuware Shared Services User/Reference Guide* for more information on the LANGUAGE command, for a list of available compilers, and for more information on this JCL and language processor options.

Postprocessor

You can use the postprocessor to process compile listings that were previously stored, or you can modify your Assembler compile JCL to pass the compiler listing to the postprocessor as described below.

- Check the assembler options.
- Change your assembler JCL to write the compiler listing (SYSPRINT) to a sequential file for use as input to the Assembler language postprocessor. A sample of the necessary JCL is shown below.

  ```
  //SYSPRINT DD UNIT=SYSDA,SPACE=(TRK,(25,20)),DCB=BLKSIZE=16093,
  //            DISP=(MOD,PASS)
  ```

  The above example is a temporary file. You can save the JCL as a permanent file to be reused as shown in the example below:

```
//SYSPRINT DD DSN=CX,STORED.LISTINGS(MEMBER)
```

- The SYSPRINT file will be LRECL=133, RECFM=FBA.
- Add SLCXCNTL sample library member CXASM as the postprocessor step following your assembler step. CXASM can be customized to meet your site's requirements.

  Note: The CWPLOAD file must be allocated using DISP=OLD or DISP=SHR.

- The object module that is processed by the postprocessor must be used as input to the linkedit step.

Refer to the *Compuware Shared Services User/Reference Guide* for more information on this JCL and language processor options.

Step 9d. Modify the JCL to Run the C Language Processor

You must perform this step if you are installing XPEDITER/TSO and you are licensed for the Compuware C language processor.

This step contains instructions for both the C language preprocessor and postprocessor. You should follow the instructions that apply to the type of processing you will be performing.

Preprocessor

SLCXCNTL sample library member CXCPRE contains sample JCL necessary to run the C language preprocessor.

Modify the compile step of your C JCL:

- On the EXEC statement, change the name of your compiler to CWPZMAIN and add the following to any existing C compiler options:

  ```
  LANGUAGE(compiler version)
  ```

  Refer to the Compuware Shared Services User/Reference Guide for a list of the compiler versions that can be specified with the LANGUAGE command. The default compiler for C is the compiler currently in use on your system.

  Note: The LANGUAGE command may be specified in the CWPPRMO command stream rather than on the EXEC statement.

- Add the CSS SLCXLOAD DSN to the STEPLIB or JOBLIB concatenation if it is not in the link list.
- Add the **CWPDDIO DD** statement to specify the name of the source listing file (DDIO file) you will be using.
- Determine the language processor options you will use as input and add them to the EXEC parameter, or add a **CWPPRMO DD** statement that points to the appropriate options. Sample options are contained in SLCXCNTL sample library member CXLPC to provide the smallest output to the source listing file (for minimizing space). If you use these options, and you will be using the C language preprocessor, add LANGUAGE(xxx) to the EXEC PARM or to the **CWPPRMO DD** statement.

Refer to the *Compuware Shared Services User/Reference Guide* for more information on this JCL and C language processor options.

Postprocessor

You can use the postprocessor to process compile listings that were previously stored, or you can modify your C compile JCL to pass the compiler listing to the postprocessor as described below.

- Check the C compiler options.
- Change the C compile JCL to write the compiler listing (SYSCPRT) to a sequential file for use as input to the C language postprocessor. A sample of the necessary JCL is shown below.

```
//SYSCPRT DD UNIT=SYSDA,SPACE=(CYL,(100,10)),
//           DCB=(RECFM=VB,BLKSIZE=16096),
//           DISP=(MOD,PASS)
```

  For more information on C language processor JCL, see the "C Language Processor" chapter in the *Compuware Shared Services User/Reference Guide*.

  The above example is a temporary file. You can save the JCL as a permanent file to be reused as shown in the example below:

```
//SYSPRINT DD DSN=CX,STORED.LISTINGS(MEMBER)
```

- The SYSPRINT file will be LRECL=133, RECFM=FBA.
- Add SLCXCNTL sample library member CXC as the postprocessor step following your C compile step. CXC can be customized to meet your site's requirements.

  Note: The CWPLOAD file must be allocated using DISP=OLD or DISP=SHR.

- The object module that is processed by the postprocessor must be used as input to the linkedit step.

See the *Compuware Shared Services User/Reference Guide* for more information on this JCL and C language processor options.

Step 10. Continue Installation of Your Compuware Product

By completing this chapter, you have finished the installation, maintenance, and customization of all required portions of ECC. Return to the appropriate manual for the Compuware product being installed and continue with the procedure provided there.

Glossary

Abend-AID XLS. The Compuware product for fault diagnosis that provides an immediate, detailed analysis of application program failures in a comprehensive report format.

Abend-AID products. A generic name for the Abend-AID product family.

Abend-AID XLS Report. (1) A set of records containing data extracted at abend time from the affected MVS region and associated control blocks that are stored in the Abend-AID XLS DDIO report file. (2) A readable report produced by CWDDSUTL at view time using the records stored in the Abend-AID XLS DDIO report file.

Abend time. The time when an abend occurs and Abend-AID products perform their analyses.

Allocation group. A set of blocks assigned to a DDIO file member when the member requires additional storage.

APF Authorization. APF can be used to restrict access to system functions and can require that all modules loaded by an authorized program be loaded from an authorized load library.All modules loaded by LMSINIT must be loaded from authorized load libraries or LMSINIT will fail. Only individuals with proper RACF authority will have permission to update authorized load libraries.

Assembler Language Processor. One of four language processors provided by Compuware. This language processor accepts Assembler output, builds sort work records, sorts and merges the records, and merges the records with the listing to produce Processor control blocks that can then be used as input to other Compuware products.

Automatic lock. Automatic locks are created whenever a member is added to a DDIO file format using the AUTODELETE=DUPS or AUTODELETE=STAGED option. The most recent member is automatically locked. See also **manual lock**.

AUTODELETE. An attribute of a DDIO file, specified during formatting of the file, that determines the action to be taken when the file becomes full and an attempt is made to add a new member to the file.

Batch File Utility. A CSS utility used to prepare and manipulate DDIO files.

C Language Processor. One of four language processors provided by Compuware. This language processor accepts C compiler output, builds sort work records, sorts and merges the records, and merges the records with the listing to produce processor control blocks that can then be used as input to other Compuware products.

CICS Abend-AID/FX. The Compuware product for fault diagnosis that provides a full range of analysis functions for managing abnormal CICS transaction terminations (abends) and CICS region outages.

COBOL Language Processor. One of four language processors provided by Compuware. This language processor accepts COBOL compiler output, builds sort work records, sorts and merges the records, and merges the records with the listing to produce processor control blocks that can then be used as input to other Compuware products.

Compuware Shared Services (CSS). A set of components used by several Compuware products to provide storage, retrieval, and maintenance for source listings and diagnostic reports.

Compuware Viewing Facility (Compuware/VF). The function of CWDDSUTL that allows the viewing, printing, locking, unlocking, and deleting of DDIO file members online.

Customer Modification Facility (CMF). This facility allows you to migrate from a previous release of CSS to CSS 7.4 or higher if you are using any site-restricted zap(s). This facility can retrofit your zap(s) to the SMP/E environment.

Customized Translation Table. An optional table provided by Compuware customers for translating non-printable characters. It is used by the batch file utility and Compuware/VF for printing or displaying lines containing storage displayed by Abend-AID products at abend time.

CWAASDUT. The batch file utility used by Abend-AID XLS shared directories and report databases. CWAASDUT allows you to manage the files used by an Abend-AID XLS shared directory and its associated abend report files. The shared directory is used to process all directory requests related to an abend report file.

CWASSECU. The name of the user-coded Security Exit program.

CWDDLPUT. The batch file utility program used by the language processor for C language, Visu-

alAge for PL/I, and Enterprise PL/I processing if using LONGNAME compiler support.

CWDDSUTL. The program for Compuware/VF and the batch file utility used by Abend-AID XLS, XPEDITER/CICS, and XPEDITER/TSO. CWDDSUTL allows you to print, delete, lock, unlock, copy, move, import, and export DDIO file. members, and to list DDIO file directory entries. It also allows you to initially format a DDIO file

CWFXSDUT. The batch file utility used by CICS Abend-AID/FX. CWFXSDUT allows you to manage the files used by a shared directory and its associated transaction databases. The shared directory is used to process all directory requests related to a transaction database.

CWPPRMO. A language processor input dataset that contains all of the language processor options.

DBMODEL. The name of an OBJECT used to define all of the specifications needed to define a new source listing database during "dynamic database creation." The information is stored in the Source Listing Shared Directory specified. It is used to build the dynamic source listing database define and format functions when the dynamic database creation process is invoked during the C Language Processor source member create.

DDIO (Dump Dataset Input Output). A Compuware proprietary file access method. The CSS language processors analyze the output from a compiler or assembler execution and store that information in a source listing DDIO file. CSS DDIO files are used to store diagnostic reports, transaction reports, and source listings. The term **DDIO file** is a generic term used to refer to datasets that store the reports and listings from Compuware products that use CSS.

DDIO File. A generic name for a report file or source listing file.

DDIO File Member. A generic name for an diagnostic report in a report file, a source listing in a source listing file, or a transaction report in a transaction database.

Directive parameter. A parameter that either specifies how the execution of a batch file utility command for existing DDIO files should be performed or changes the defaults for certain ddnames.

Directory entry. A record in the fixed portion of the DDIO file (Directory) that contains information specific to a member. Each member has a corresponding directory entry. The number of directory entries is specified during formatting of the DDIO file and cannot be changed without reformatting the file.

DIRX Report. A report produced by the DIRX command of the batch file utility.

Distributed Viewing Support (DVS). Allows Abend-AID XLS users (version 9.0 and above) to view both merged and base reports, where either the base report or listing file resides on a remote system. Abend-AID XLS users can have Extended Language Support across remote MVS systems that don't share DASD.

DYNCREATE. A term used to denote the "dynamic database creation" process that can be invoked when using the C Language Processor shared directory and source listing databases for its compiler output (CWPDDIO). A DBMODEL must first be defined in the desired Source Listing Shared Directory using the batch utility CWDDLPUT with DYNCREATE=YES specified to allow invoking of the dynamic database creation process. The CWPDDIO must specify the Source Listing Shared Directory DSN name. If not enough space is found in any of the attached source listing databases, and DYNCREATE=YES, then the dynamic database creation process is invoked, the new source listing database is created and attached, and the new source member is created.

Enhanced Listing. A convenient source of quick reference information and program documentation that merges DMAP and CLIST information, in addition to error and diagnostic messages, with a COBOL source listing.

Enterprise Common Components (ECC). ECC is the packaging method for the following Compuware facilities:

- License Management System (LMS)
- Compuware Shared Services (CSS)
- Distributed Viewing Support (DVS)

Entry. A generic name for a transaction diagnostic report in a transaction database for CICS Abend-AID/FX.

Formatting a DDIO File. The preparation of an allocated file to be used as a DDIO file.

Full DDIO File. A DDIO file for which all allocation groups and/or all directory entries are allocated to members.

Host Communications Interface (Base Services/HCI). A facility that provides connectivity between mainframe-based programmer productivity software and peer node software running on other platforms in a network. It allows application programs to communicate over any one of several

protocols without knowledge of which protocol is in use at any given time.

Installation JCL Customization Facility. A CSS facility consisting of a set of screens that prompt you to enter installation information. This information is used to build the jobs necessary to perform the CSS SMP/E installation.

Language Processor (LP). A processor that converts assembler or compiler output into input for other Compuware products.

License Administration Utility (LAU). The LAU is the license administration control center for your organization's IT professional who is responsible for managing your access to Compuware products. The LAU is an ISPF application that enables

- the creation of a License File
- the import of License Certificates into a License File
- the maintenance and export of a License File
- the reporting and analysis of your License File and License Certificate information in virtual storage.

Your organization's License Administrator will also set up License Management System parameters and system operation options using the LAU.

License Certificate. English-like readable electronic records that contain a portion of the information from your license agreement for a product release's use at a particular site. The License Certificate is used to update your License File.

License File. Dataset containing imported License Certificate information for all licensed releases of Compuware products. This file is in ASCII text format.

License Management System (LMS). Facility that enables you to centrally administer Compuware product License Certificates and manage access to Compuware products at your site. The LMS includes several components that together enable you to establish, maintain, diagnose, and upgrade access to the Compuware products licensed by your enterprise. The LMS replaces the customer profile utility provided with earlier versions of Compuware mainframe products.

Verification Report (LMVERIFY). Displays the result of a verification program that checks out each product version in cache. This data is exactly what a product would receive when it runs. Running this report will prove that the License Management System environment is properly set up for the customer.

Line command. A command entered next to the line to be processed. A line command is executed only for the specified member.

Locked member. A DDIO file member that was manually locked using the batch file utility or Compuware/VF, or automatically locked as a result of AUTODELETE=DUPS or AUTODELETE=STAGED processing. A manually locked member is identified by an M in the directory report or directory screen while an automatically locked member is identified by an L in the directory report or directory screen. When manually locked, a member cannot be automatically deleted when the DDIO file becomes full.

Manual lock. A member that was locked using the **L** line command in Compuware/VF or by using the LOCK batch command. Members that have been manually locked retain their locks during the autodeletion process. See also **automatic lock**.

Oldest member. (1) For report files, the diagnostic report with the earliest abend date and time. (2) For source listing files, the program with the earliest compile date and time.

PL/I Language Processor. One of four language processors provided by Compuware. This language processor accepts PL/I compiler output, builds sort work records and an incore symbol table of all of the identifiers, and produces Processor control blocks that can then be used as input to other Compuware products.

Primary Command. A command entered in the COMMAND INPUT field.

Problem Documentation Utility. A CSS utility that captures the documentation needed by Compuware for resolving technical issues and problems. This utility allows you to collect the documentation — such as copybooks, SYSPRINT, and DDIO members — that CSS Technical Support requires for problem resolution.

Report File. A file containing diagnostic reports that is accessed by DDIO.

Security Exit Program. A user-coded program that allows or denies execution of a requested command for a selected member by a specific user against particular DDIO file(s).

Selective Parameter. A parameter that specifies the members of a DDIO file that will be used during execution of a batch file utility command for existing DDIO files.

Shared Directory. A variable-length record VSAM RRDS that contains CICS Abend-AID/FX directory records for each region and transaction

dump known to a server, or contains Source Listing Shared Directory records necessary to process Source Listing Database members.

SMP/E. System Modification Program/Extended is IBM's standard facility for maintaining software modifications on MVS.

Source Listing. A compiled listing and other information about a program stored in a source listing file.

Source Listing Database. A specially formatted source listing file, owned and managed by a shared directory, that is used by the C Language Processor for C LONGNAME support.

Source Listing File. A file containing source listings that is accessed by DDIO.

Strobe. The STROBE MVS Application Performance Measurement System is a product that determines where and how application time is spent in online regions and batch processing programs and how system resources are used. STROBE collects several types of data as it tracks activity within an MVS environment and produces a collection of reports that helps you determine where to revise applications to improve their performance.

Suppressed statements. Statements that are not displayed in the compiler listing. Suppressed statements in a DDIO source member contain an internal flag that the XPEDITER/TSO product can use to display or suppress source statements using the GEN command.

Transaction Database. A DDIO file, containing transaction reports, that is managed by a shared directory in CICS Abend-AID/FX. Multiple transaction reports are stored in a single transaction database.

Transaction Report. A CICS Abend-AID/FX report created from CICS transaction terminations that is stored in a DDIO file.

Unlocked Member. A DDIO file member that is not currently locked. The locked member can be unlocked either manually, using the batch file utility or Compuware/VF, or automatically as a result of AUTODELETE=DUPS or AUTODELETE=STAGED processing. An unlocked member in a full DDIO file formatted with AUTODELETE=YES, AUTODELETE=DUPS, or AUTODELETE=STAGED can be automatically deleted when another member is added to the file.

View time. The time when an diagnostic report or source listing is presented in a readable format for viewing online or printing.

Index

Special Characters

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

W

X